AD-A055 120 DAYTON UNIV OHIO SCHOOL OF ENGINEERING F/G 15/5

ANALYSIS OF INVENTORY/RESUPPLY SUB-SYSTEM MODELS IN OPTIMAL MAT--ETC(U)

APR 78 A Z KOVACS AFOSR-77-3266

UNCLASSIFIED UDR-TR-78-40 NL

1 OF 2

AD A055120

AFOSR-TR- 78-0904

FOR FURTHER TRAN

(2)

UDR-TR-78-40
UDSE-TR-78-04

A055120

ANALYSIS OF INVENTORY/RESUPPLY SUB-SYSTEM MODELS IN OPTIMAL MATERIEL DISTRIBUTION SYSTEM DESIGN

Arthur Z. Kovacs

AD No.
DDC FILE COPY

April 1978

School of Engineering
UNIVERSITY OF DAYTON
DAYTON, OHIO

14

UDR-TR-78-40,
UDSE-TR-78-04

6

ANALYSIS OF INVENTORY/RESUPPLY SUB-SYSTEM MODELS IN OPTIMAL MATERIEL DISTRIBUTION SYSTEM DESIGN.

9 Final Rep[redacted]t.,

15 ✓AFOSR-77-3266

16 2304

17 D9

by

10 Arthur Z./Kovacs

April 30, 1978

11 30 Apr 78

12 157p.

Engineering Services

School of Engineering

University of Dayton

Dayton, Ohio 45469

for

Department of the Air Force

Air Force Office of Scientific Research

Bolling Air Force Base, D.C. 20332

Directorate of Mathematical and Information Sciences

409 377

AH

FOREWARD

The work reported in this document was accomplished by Dr. Arthur Z. Kovacs under Grant AFOSR-77-3266 awarded to the University of Dayton by the Department of the Air Force, Air Force Office of Scientific Research, Directorate of Mathematical and Information Sciences, Bolling Air Force Base. The author thanks the Air Force Office of Scientific Research which made this work possible and the Air Force Logistics Command which introduced him to this problem area during the 1976 United States Air Force/American Society for Engineering Education Summer Faculty Research Program at Wright-Patterson Air Force Base, Ohio. In particular, he thanks Douglas Allen, a student in the Department of Mechanical Engineering at the University of Dayton, for his excellent work in writing the computer programs used to perform the analyses described in this report. Finally, special thanks is given to Dr. David C. Kraft, who as Dean of the School of Engineering, provided valuable support toward the completion of this project.

ABSTRACT

The purpose of the Air Force freight distribution system is to meet spare parts demand requirements at minimum cost. Budgetary constraints ~~have~~ suggested that total expected backorder level for items at user installations be minimized, subject to a given dollar expenditure level for inventory investment. LOGAIR (a dedicated Air Force air transport service) is a major transportation sub-system to support spare parts delivery requirements of users of high priority items. A two-echelon inventory system for spare parts delivery exists with centralized, specialized inventories at ~~the~~ Air Logistics Centers (ALCs) and decentralized, broad profile inventories located at user installations. LOGAIR provides transport with low order and ship time to reduce resupply time in the maintenance of inventory safety levels at user bases. A systems approach is used to formulate a cost/benefit model, ~~which~~ is formulated recognizes the impact of the Air Force resupply system upon ~~the~~ total spare parts distribution ~~system~~ in terms of total inventory investment level, total system cost, and backorder level. Given a total expenditure level available for allocation between inventory investment and transportation, the problem is to determine the optimal fractional allocation to be made to transportation (the remaining fraction to be allocated to inventory investment) such that total expected backorder level is minimized. A conceptual framework for trade-off analysis for minimizing total system cost in terms of inventory investment level and resupply time level for a given backorder level is presented. This conceptual framework also allows for the minimization of backorder level in terms of inventory investment level and resupply time level for a given total dollar expenditure level. Queueing models exhibiting various demand and resupply processes are explored and compared to determine the impact of inventory investment level and resupply time level upon backorder level. Specific solution procedures are developed for and are applied to the trade-off analyses mentioned above.

TABLE OF CONTENTS

SECTION 1
INTRODUCTION

The purpose of the Air Force freight distribution system is to meet spare parts demand requirements of user installations at minimum cost. Budgetary constraints have suggested that total expected backorder level for items at user installations be minimized subject to a given dollar expenditure level for inventory investment. LOGAIR (a dedicated Air Force air transport service) is a major transportation sub-system to support spare parts delivery requirements to users of high priority items. A two-echelon inventory support system for spare parts delivery exists with centralized, specialized inventories at the Air Logistics Centers (ALCs) and decentralized, broad profile inventories located at user installations. LOGAIR provides transport with low order and ship time to reduce resupply time in the maintenance of inventory safety levels at the user bases.

The total systems problem is to determine an optimal logistics support system considering the cost trades-off which are possible among inventory, repair and maintenance, and transportation sub-systems as well as management policies regarding procurement and level of service. Within this framework, the transportation system is to be developed to utilize available transportation modes in servicing requirements in a timely, cost-effective manner. Any mode which is dedicated to Air Force needs (LOGAIR, truck) must be defined in detail to include route specification, vehicle schedules, and managerial procedures. Thus, a systems approach is proposed to explore formulation of a cost/benefit model which recognizes the impact of the Air Force resupply system upon the total spare parts distribution system in terms of total inventory investment level, total system cost, and backorder level. This approach recognizes that the purpose of expenditures for transportation is to reduce resupply time in the support of inventory policy designed to meet user needs.

Implicit in the approach taken by any inventory model is the assumption of the existence of a transportation system capable of delivery of an order and ship time (O&ST) profile which makes up part of the resupply time required for input to the model. The Air Force freight transportation system is supposed to provide timely delivery of inventory items to users from the supply depots and to ship items from the user installations back to the depots for repair. A transportation system with low average order and ship time levels should yield correspondingly low levels of required inventory investment at a cost which is small compared to any resultant savings in inventory investment. Another purpose of the transportation system is to minimize resupply time in the event of a stockout.

The problem then becomes:

> Given a total expenditure level available for allocation between inventory investment and transportation, what is the optimal fractional allocation to be made to transportation (the remaining fraction to be allocated to inventory investment) such that the total expected back-order level at the user installations is minimized?

In this allocation, it is assumed that expenditures are to be expended in an optimal fashion. Inventory models which can treat consumable as well as recoverable items are to optimally allocate total inventory investment among supply depots and user installations subject to a given input O&ST profile. A transportation network and mode selection model can be required to optimally utilize a given transportation expenditure level to yield an optimum O&ST profile that minimizes average O&ST level experienced within the network. Or, conversely, a transportation network and optimal mode selection model might be designed to meet specific O&ST standards for all shipments in such a fashion as to minimize total transportation cost.

The above approach ignores maintenance and repair costs and other costs associated with other segments in the resupply cycle. Expenditure

levels in these areas are assumed to be optimal in relation to the total expenditure level allocated for inventory investment and transportation. In principle, a methodology could be developed to include optimal allocation of resources to other segments of the resupply cycle.

SECTION 2

GENERAL MATHEMATICAL STATEMENT OF PROBLEM

Let $[N_B]$ be the expected number of backorders for a given single investment (recoverable) item. $[N_B]$ can be written

$$[N_B] = [N_B(N, \rho)] \quad , \tag{2.1}$$

where

N = total number of recoverable items including those installed,

ρ = $\lambda/\mu = \lambda\tau_\mu$ = ratio of item demand rate to item resupply rate,

λ = $\frac{1}{\tau_\lambda}$ = item demand rate,

τ_λ = mean time between demands,

μ = $\frac{1}{\tau_\mu}$ = resupply rate,

τ_μ = mean resupply time.

The total cost associated with maintaining the inventory/resupply system can be written

$$C = C(N, \rho) \quad . \tag{2.2}$$

Suppose it is desired to maintain a given "safe" backorder level so that

$$[N_B] = N_B^{(o)} \quad , \tag{2.3}$$

where $N_B^{(o)}$ is a specified constant, and the objective is to minimize system cost. Then the problem becomes

minimize

$$C = C(N, \rho)$$

subject to

$$[N_B(N, \rho)] = N_B^{(o)} \tag{2.4}$$

To facilitate discussion, assume N and ρ to be continuous variables; the generalization to the case where N is discrete can be achieved through usage of first difference operations for discrete variables. Thus,

$$\delta C = \left(\frac{\partial C}{\partial N}\right)_\rho \delta N + \left(\frac{\partial C}{\partial \rho}\right)_N \delta \rho = 0 \quad ; \tag{2.5}$$

$$\delta[N_B] = \left(\frac{\partial[N_B]}{\partial N}\right)_\rho \delta N + \left(\frac{\partial[N_B]}{\partial \rho}\right)_N \delta \rho = 0 \quad ; \tag{2.6}$$

$$\delta N = -\frac{\left(\frac{\partial[N_B]}{\partial \rho}\right)_N}{\left(\frac{\partial[N_B]}{\partial N}\right)_\rho} \delta \rho \quad . \tag{2.7}$$

For physical systems $\left(\frac{\partial[N_B]}{\partial \rho}\right)_N > 0$, $\left(\frac{\partial[N_B]}{\partial N}\right)_\rho < 0$, $\left(\frac{\partial C}{\partial N}\right)_\rho > 0$, and $\left(\frac{\partial C}{\partial \rho}\right)_N < 0$.

Note that

$$\left(\frac{\partial N}{\partial \rho}\right)_{[N_B]} = -\frac{\left(\frac{\partial[N_B]}{\partial \rho}\right)_N}{\left(\frac{\partial[N_B]}{\partial N}\right)_\rho} > 0 \quad , \tag{2.8}$$

and

$$\left(\frac{\partial N}{\partial \rho}\right)_C = -\frac{\left(\frac{\partial C}{\partial \rho}\right)_N}{\left(\frac{\partial C}{\partial N}\right)_\rho} > 0 \quad . \tag{2.9}$$

Thus,

$$\delta N = +\left(\frac{\partial N}{\partial \rho}\right)_{[N_B]} \delta\rho \tag{2.10}$$

and

$$\delta N = +\left(\frac{\partial N}{\partial \rho}\right)_{C} \delta\rho \quad , \tag{2.11}$$

which implies

$$f(N,\rho) \equiv \left(\frac{\partial N}{\partial \rho}\right)_{C} - \left(\frac{\partial N}{\partial \rho}\right)_{[N_B]} = 0 \quad . \tag{2.12}$$

If equation (2.12) and equation (2.4) are solved simultaneously, then solution values of (N, ρ) are obtained which minimize system cost $C(N, \rho)$. If, on the other hand, backorder level is to be minimized subject to a given total dollar investment level so that

$$C(N,\rho) = C_o \tag{2.13}$$

then equation (2.12) and equation (2.13) are solved simultaneously to obtain the desired values of (N, ρ).

In order to determine the optimal (minimum) system expenditure level to be allocated between inventory investment level (N) and resupply time level ($\tau_\mu = \rho\tau_\lambda$), the following functions must be determined:

$$[N_B] = [N_B(N,\rho)] > 0 \quad , \tag{2.14}$$

$$g_1(N,\rho) = \left(\frac{\partial [N_B]}{\partial N}\right)_{\rho} < 0 \quad , \tag{2.15}$$

$$g_2(N,\rho) = \left(\frac{\partial [N_B]}{\partial \rho}\right)_{N} > 0 \quad , \tag{2.16}$$

$$g_3(N,\rho) = \left(\frac{\partial N}{\partial \rho}\right)_{[N_B]} > 0 \quad ; \qquad (2.17)$$

$$C = C(N,\rho) > 0 \quad , \qquad (2.18)$$

$$g_4(N,\rho) = \left(\frac{\partial C}{\partial N}\right)_{\rho} > 0 \quad , \qquad (2.19)$$

$$g_5(N,\rho) = \left(\frac{\partial C}{\partial \rho}\right)_{N} < 0 \quad , \qquad (2.20)$$

$$g_6(N,\rho) = \left(\frac{\partial N}{\partial \rho}\right)_{C} > 0 \quad . \qquad (2.21)$$

The determination of relations (2.14) through (2.17) (STUDY NO. 1) is a study of the impact of inventory investment level (N) and resupply time level ($\tau_\mu = \rho\tau_\lambda$) upon backorder level. The determination of relations (2.18) through (2.21) (STUDY NO. 2) is a study of the impact of inventory investment level and resupply time level upon total system cost. The results of STUDY NO. 1 and STUDY NO. 2 can be used to determine optimum allocation of resources between inventory investment level and resupply time level as outlined above.

Note that the resupply time τ_μ is given by

τ_μ = (repair time) + (transportation time) + (administrative time for processing shipments).

If the subject of interest is the impact of transportation time upon total system cost and required inventory investment level, then the cost function

$C(N, \rho)$ need only include costs associated with inventory investment level and transportation ship time level, where a differential change in ρ ($\delta\rho = \lambda\delta\tau_{\mu}$) is considered to be due to a differential change in transportation ship time level.

The above problem formulation is generalized to include base level repair and depot level repair (which results in the inclusion of transportation time as part of the resupply time), the interaction of more than one base with a single depot, and can be modified to include the interactions due to movement of many investment and consumable items in the transportation pipelines among user bases and supply depots. If desired, the impact of administrative processing time and/or of repair time upon total system cost in relation to resulting required inventory investment level can also be determined using the conceptual procedure outlined above.

SECTION 3

DEMAND AND RESUPPLY PROCESSES IN INVENTORY MODELING

When dealing with failures of units in operation it is reasonable to assume that

1. The probability for a given unit in operation to fail in any particular operational time interval of arbitrary size is the same for all identical operational units;
2. The fact that a given unit in operation has failed in a given operational time interval does not affect the probability that other identical units in operation may fail in the same operational time interval (independence of failures);
3. The probability for a given unit in operation to fail during a given operational time interval is the same for all operational time intervals of equal arbitrary size (the mean operational time to failure is long compared to the total operational time period of observation) no matter when in time the failure takes place (time independence of failure rate).

The above conditions are necessary and sufficient to define the probability for failure of a given unit in operation during a given finite operational time interval t. Thus for small Δt, let

$$\lambda \Delta t = \text{probability for failure of an item in operation during the operational time interval } \Delta t \,, \tag{3.1}$$

and

$$1 - \lambda \Delta t = \text{probability of a given unit in operation for not failing during the operational time interval } \Delta t \quad . \tag{3.2}$$

If P(t) is the probability that a given unit in operation does not fail during a finite operational time interval t, then

$$P(t + \Delta t) = (1 - \lambda \Delta t)\, P(t) \quad , \tag{3.3}$$

so that

$$\frac{dP(t)}{dt} = \lim_{\Delta t \to 0} \frac{P(t + \Delta t) - P(t)}{\Delta t} = -\lambda P(t) \quad ; \qquad (3.4)$$

$$\frac{dP(t)}{dt} = -\lambda e^{-\lambda t} \quad ; \qquad (3.5)$$

$$P(t) = e^{-\lambda t} \quad . \qquad (3.6)$$

The above results apply to a single unit in operation and are independent of the number of units in operation in the system. If, in addition, one requires that

4. The total number of units in operation and the total number of equal arbitrarily chosen operational time intervals are large (making statistical averages significant),

then the failure process is Poisson. If the probability for observing m failures in an operational time t is given by $P_m(t)$, then

$$P_m(t + \Delta t) = P_m(t)(1 - \lambda\Delta t) + P_{m-1}(t)\lambda\Delta t \quad , \qquad \infty > m \geq 1 \quad ; \qquad (3.7)$$

$$P_0(t + \Delta t) = P_0(t)(1 - \lambda\Delta t) \quad , \qquad m = 0 \quad . \qquad (3.8)$$

Thus when $\Delta t \to 0$, one obtains

$$\frac{\partial P_0(t)}{\partial t} = -\lambda P_0(t) \quad , \qquad m = 0 \; ; \qquad (3.9)$$

$$\frac{\partial P_m(t)}{\partial t} = -\lambda P_m(t) + \lambda P_{m-1}(t) \quad , \qquad \infty > m \geq 1 \; . \qquad (3.10)$$

Application of standard techniques obtains

$$P_m(t) = \frac{(\lambda t)^m e^{-\lambda t}}{m!} \quad , \qquad 0 \leq m < \infty \quad . \tag{3.11}$$

It is observed that the Poisson process only applies if the number of units in operation is very large, or if as soon as a unit fails, it is immediately replaced (resupply time ≡ 0) by an operational unit. If no backorders occur (i.e., there always exists sufficient servicable units in stock available for necessary immediate replacement of failed units), then the Poisson failure process applies; when backorders are allowed to exist for time intervals of significant duration, or if replacement time is significant, then the Poisson failure process does not apply. Thus while exponential operational failure times may be assumed, in general a Poisson failure process cannot be assumed without careful investigation into the nature of actual resupply times.

Assume that probability to complete a service on a given unit satisfies the following conditions.

1. The probability for a given unit in service to complete service in a particular time interval is the same for all identical units in service being served by identical servers;
2. The fact that a given unit in service has completed service in a given time interval does not affect the probability that other units in service in other servers may complete service in the same time interval (independence of service completion times);
3. The probability for a given unit in service to complete service during a given time interval is the same for all time intervals of equal size (the mean time to complete service is long compared to the total period of observation) no matter when the service completion takes place (time independence of service rate);
4. The number of units in service does not affect the probability for completion of service for an item in service.

The above conditions are necessary and sufficient to define the probability for completion of service for an item in service during an arbitrary given time interval t. Thus, for small Δt let

$$\mu\Delta t = \text{probability for completing service for an item in service during } \Delta t\,, \tag{3.12}$$

$$1 - \mu\Delta t = \text{probability for not completing service for an item in service during } \Delta t. \tag{3.13}$$

An argument similar to that given above for failures obtains exponentially distributed service times. (Note that if the number of units awaiting service for a given single server were always non-zero, then the number completing service in an arbitrary time interval would be Poisson distributed.)

In applying servers with exponential service times in modeling the resupply process, the nature of the server configuration is extremely important. A few possible models for resupply are given below.

If a single level resupply system function is assumed, then resupply can be described by a singly indexed resupply state so that

$$\mu_n = n\mu \,, \qquad 0 \leq n \leq n_{max} \,, \tag{3.14}$$

$$\mu_n = n_{max}\mu \,, \qquad N \geq n \geq n_{max} \,, \tag{3.15}$$

$$1 \leq n_{max} \leq N \,, \tag{3.16}$$

where

μ_n = resupply rate when n units are in resupply,

μ = constant = resupply rate when a single unit is in resupply,

n_{max} = number of identical parallel servers assumed,

N = total number of units in the system (including installed, stock, and resupply items).

The single index for resupply is n, the number of units in resupply. Note that when $n_{max} = 1$, then a single server model is assumed, and when $n_{max} > 1$, then a multiple server model is assumed. If n_{max} is chosen so that $n_{max} = N$, then no service queue is formed; if $n_{max} < N$, then service queues do form.

For a dual level resupply system (two level resupply; e.g., base level and depot level) two server models may be postulated. The first is described by a resupply state with a single index so that

$$\mu_n = n\mu \quad , \qquad 0 \leq n \leq n_{max} \quad , \tag{3.17}$$

$$\mu_n = n_{max}\mu, \qquad n_{max} \leq n \leq N \quad , \tag{3.18}$$

$$1 \leq n_{max} \leq N \quad ; \tag{3.19}$$

$$\mu \equiv f_1\mu_1 + f_2\mu_2 \quad ; \tag{3.20}$$

where

f_1 = probability for level 1 resupply (e.g., base level resupply),

f_2 = probability for level 2 resupply (e.g., depot level resupply),

$f_1 + f_2$ = 1,

μ_1 = resupply rate for one unit in level 1 resupply,

μ_2 = resupply rate for one unit in level 2 resupply.

Again the single index for resupply is n, the total number of units in resupply. The form for the above dual level resupply is identical to that for single level resupply except that the effective service rate μ is given in terms of a weighted combination of μ_1 and μ_2.

A second approach to model a dual level resupply system is to use a two index resupply state. The description of resupply rates is given by

$$\left.\begin{aligned} \mu_\alpha^{(1)} &= \alpha\mu_1 \quad , & 0 \leq \alpha \leq \alpha_{max} \quad , \\ \mu_\alpha^{(1)} &= \alpha_{max}\mu_1 \quad , & \alpha_{max} \leq \alpha \leq N \quad ; \end{aligned}\right. \tag{3.21}$$

$$\mu_\beta^{(2)} = \beta\mu_2 \quad , \qquad 0 \le \beta \le \beta_{max} \quad ,$$

$$\mu_\beta^{(2)} = \beta_{max}\mu_2 \quad , \qquad \beta_{max} \le \beta \le N \quad ; \qquad (3.21)$$

$$\mu \equiv f_1\mu_1 + f_2\mu_2 \ , \qquad f_1 + f_2 = 1 \quad ;$$

where

$\mu_\alpha^{(1)}$ = resupply rate for level 1 resupply when α units are in resupply at level 1,

$\mu_\beta^{(2)}$ = resupply rate for level 2 resupply when β units are in resupply at level 2,

α = number of units in resupply at level 1,

β = number of units in resupply at level 2,

$\alpha + \beta$ = n = total number of units in resupply, $0 \le \alpha + \beta = n \le N$,

f_1 = probability for level 1 resupply,

f_2 = probability for level 2 resupply,

α_{max} = assigned number of identical parallel servers at level 1, $1 \le \alpha_{max} \le N$,

β_{max} = assigned number of identical parallel servers at level 2, $1 \le \beta_{max} \le N$,

μ_1 = resupply rate at level 1 when one unit is in resupply at level 1,

μ_2 = resupply rate at level 2 when one unit is in resupply at level 2,

N = total number of units in the system (including installed items as well as items in resupply and in stock).

The two index resupply state is labeled by (α, β), the number of resupply items at level 1 and the number of resupply items at level 2, respectively.

In the single index resupply state models given above, the arrival rate for resupply (failure rate) can be modeled by

$$\lambda_m = m_0\lambda \quad , \qquad m_0 \le m \le N \quad ; \qquad (3.22)$$

$$\lambda_m = m\lambda \qquad , \qquad 0 \le m \le m_0 \quad ; \tag{3.23}$$

or equivalently by

$$\lambda_m = m_0\lambda \qquad , \qquad 0 \le n \le N - m_0 \quad ; \tag{3.24}$$

$$\lambda_m = (N - n)\lambda \quad , \quad N - m_0 + 1 \le n \le N \; ; \tag{3.25}$$

where

n = number of units in resupply,

λ_m = resupply arrival rate when there are m units installed,

λ = resupply arrival rate when there is one unit installed,

m_0 = minimum number of installed items resulting in meeting total operational requirements with installed units operating at full capability (when $m > m_0$, then total operational requirements are met with installed items operating at less than full capability; when $m < m_0$ total operational requirements are not met with installed items operating at full capability),

N = total number of units in system.

In the two index resupply state model, the arrival process can be described by

$$\lambda_m^{(1)} = f_1 m_0 \lambda \quad , \quad m_0 \le m \le N \; ,$$

$$\lambda_m^{(2)} = f_2 m_0 \lambda \quad , \quad m_0 \le m \le N \; ,$$

$$\lambda_m^{(1)} = f_1 m \lambda \quad , \quad 0 \le m \le m_0 , \tag{3.26}$$

$$\lambda_m^{(2)} = f_2 m \lambda \quad , \quad 0 \le m \le m_0 ,$$

$$f_1 + f_2 = 1 \qquad ,$$

where

λ = failure rate when one unit is installed,

$\lambda_m^{(1)}$ = resupply arrival rate for level 1 when m units are installed,

$\lambda_m^{(2)}$ = resupply arrival rate for level 2 when m units are installed,

f_1 = probability for level 1 resupply,

f_2 = probability for level 2 resupply,

m = number of installed units,

m_0 = minimum number of installed units resulting in meeting total operational requirements.

Define $\eta_n^{(1)}$ to be the resupply arrival rate at level 1 when the total number in resupply is n and $\eta_n^{(2)}$ to be the resupply arrival rate at level 2 when the total number in resupply is n, $0 \leq n = \alpha + \beta \leq N$. Then

$$\eta_n^{(1)} = f_1 m_0 \lambda \quad , \qquad 0 \leq n \leq N - m_0 \; ; \tag{3.27}$$

$$\eta_n^{(2)} = f_2 m_0 \lambda \quad , \qquad 0 \leq n \leq N - m_0 \; ; \tag{3.28}$$

$$\eta_n^{(1)} = f_1 (N - n) \lambda \; , \quad N - m_0 + 1 \leq n \leq N \; ; \tag{3.29}$$

$$\eta_n^{(2)} = f_2 (N - n) \lambda \; , \quad N - m_0 + 1 \leq n \leq N \; . \tag{3.30}$$

The state transition probabilities for a singly indexed resupply state are given for $n_{max} = N$ and $\mu_n = n\mu$, $0 \leq n \leq N$ (to first order in Δt) by

$$P_0(t + \Delta t) = P_0(t)[1 - m_0 \lambda \Delta t] + P_1(t)[\mu \Delta t] \quad , \qquad n = 0 \quad ; \tag{3.31}$$

$$P_n(t + \Delta t) = P_n(t)[1 - m_0\lambda\Delta t - n\mu\Delta t] + P_{n+1}(t)[(n + 1)\mu\Delta t] + P_{n-1}(t)[m_0\lambda\Delta t] ,$$

$$0 < n \leq N - m_0 ; \qquad (3.32)$$

$$P_n(t + \Delta t) = P_n(t)[1 - (N - n)\lambda\Delta t - n\mu\Delta t] + P_{n+1}(t)[(n + 1)\mu\Delta t]$$

$$+ P_{n-1}(t)[(N - n + 1)\lambda\Delta t] ,$$

$$N - m_0 + 1 \leq n < N ; \qquad (3.33)$$

$$P_N(t + \Delta t) = P_N(t)[1 - N\mu\Delta t] + P_{N-1}(t)[\lambda\Delta t] , \qquad n = N . \qquad (3.34)$$

The differential equations which result when $\Delta t \to 0$ and the corresponding steady-state solution are given in the section entitled "Conservative Parallel Server Inventory Model."

When $\mu_n = \mu$ for $n \geq 1$ and $\mu_n = 0$ for $n = 0$, the state transition probabilities for a singly indexed resupply state are given (to first order in Δt) by

$$P_0(t + \Delta t) = P_0(t)[1 - m_0\lambda\Delta t] + P_1(t)[\mu\Delta t] , \qquad n = 0 ; \qquad (3.35)$$

$$P_n(t + \Delta t) = P_n(t)[1 - m_0\lambda\Delta t - \mu\Delta t] + P_{n+1}(t)[\mu\Delta t] + P_{n-1}(t)[m_0\lambda\Delta t] ,$$

$$0 < n \leq N - m_0 ; \qquad (3.36)$$

$$P_n(t + \Delta t) = P_n(t)[1 - (N - n)\lambda\Delta t - \mu\Delta t] + P_{n+1}(t)[\mu\Delta t]$$

$$+ P_{n-1}(t)[(N - n + 1)\lambda\Delta t] ,$$

$$N - m_0 + 1 \leq n < N ; \qquad (3.37)$$

$$P_N(t + \Delta t) = P_N(t)[1 - \mu\Delta t] + P_{N-1}(t)[\lambda\Delta t] \quad , \qquad n = N . \tag{3.38}$$

The differential equations which result when $\Delta t \to 0$ and the corresponding steady-state solution are given in the section entitled "Conservative Single Server Inventory Model."

The two index resupply state model is described (to first order in Δt) by

$$P_{\alpha,\beta}(t + \Delta t) = P_{\alpha,\beta}(t)[1 - \eta^{(1)}_{\alpha+\beta}\Delta t - \eta^{(2)}_{\alpha+\beta}\Delta t - \mu^{(1)}_{\alpha}\Delta t - \mu^{(2)}_{\beta}\Delta t]$$

$$+ P_{\alpha+1,\beta}(t)[\mu^{(1)}_{\alpha+1}\Delta t] + P_{\alpha-1,\beta}(t)[\eta^{(1)}_{\alpha+\beta-1}\Delta t]$$

$$+ P_{\alpha,\beta+1}(t)[\mu^{(2)}_{\beta+1}\Delta t] + P_{\alpha,\beta-1}(t)[\eta^{(2)}_{\alpha+\beta-1}\Delta t], \tag{3.39}$$

$$0 \leq \alpha + \beta = n \leq N ,$$

where

$$P_{\alpha,\beta}(t) \equiv 0 \text{ when } \alpha < 0 \text{ or when } \beta < 0.$$

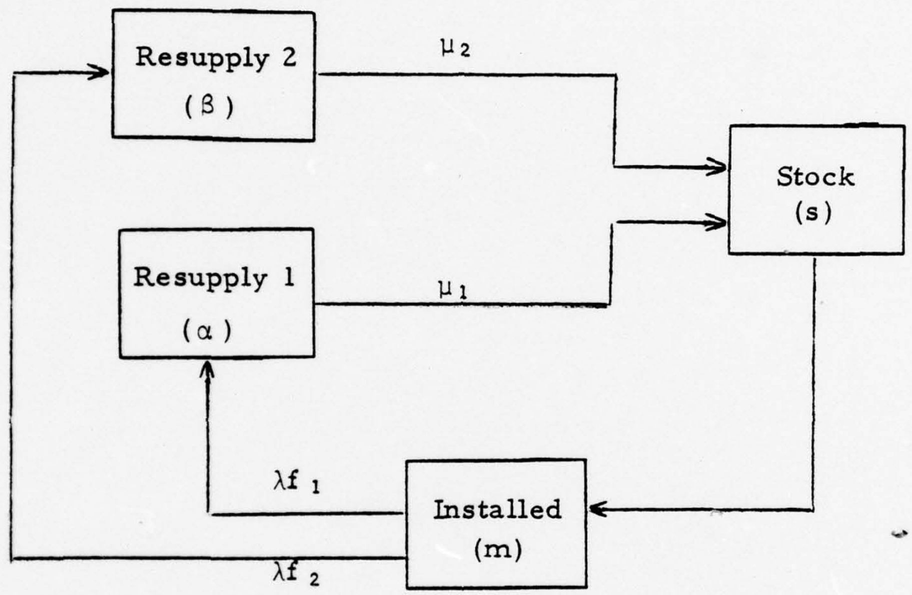
The differential equations obtained when $\Delta t \to 0$ are

$$\frac{\partial P_{\alpha,\beta}(t)}{\partial t} = -[(\eta^{(1)}_{\alpha+\beta} + \eta^{(2)}_{\alpha+\beta}) + (\mu^{(1)}_{\alpha} + \mu^{(2)}_{\beta})] P_{\alpha,\beta}(t) + \mu^{(1)}_{\alpha+1} P_{\alpha+1,\beta}(t)$$

$$+ \mu^{(2)}_{\beta+1} P_{\alpha,\beta+1}(t) + \eta^{(1)}_{\alpha+\beta-1} P_{\alpha-1,\beta}(t) + \eta^{(2)}_{\alpha+\beta-1} P_{\alpha,\beta-1}(t) , \tag{3.40}$$

$$0 \leq \alpha + \beta = n \leq N.$$

The two index resupply state system described above is one that most closely describes the two-echelon inventory system currently adopted by Air Force management [88, 89, 104] and is of major usefulness in the trade-off analysis described earlier in this document. Below is diagrammed the conservative two level parallel server inventory model, and a mathematical treatment showing the equivalence of this model to the singly indexed parallel server model follows.



CONSERVATIVE TWO LEVEL PARALLEL SERVER INVENTORY MODEL

Define

$$P_n(t) \equiv \sum_{\alpha=0}^{n} P_{\alpha,n-\alpha}(t) \quad , \tag{3.41}$$

and observe that

$$P_{\alpha,n-\alpha}(t) = f_1^{\alpha} f_2^{n-\alpha} \binom{n}{\alpha} P_n(t) \quad , \qquad 0 \leq \alpha \leq n, \quad 0 \leq n \leq N, \tag{3.42}$$

where $P_n^{(t)}$ = the probability that there are n units in resupply at time t,

$$n = \alpha + \beta, \text{ and} \binom{n}{\alpha} = \frac{n!}{(n-\alpha)!\alpha!} \, .$$

Further,

$$\begin{aligned} \eta_n^{(1)} + \eta_n^{(2)} &= m_0\lambda \equiv \lambda_n \quad , \qquad 0 \le n \le N-m_0 \quad , \\ \eta_n^{(1)} + \eta_n^{(2)} &= (N-n)\lambda \equiv \lambda_n \, , \qquad N-m_0 + 1 \le n \le N \quad . \end{aligned} \tag{3.43}$$

Therefore,

$$\eta_n^{(1)} + \eta_n^{(2)} \equiv \lambda_n \quad , \qquad 0 \le n \le N \quad , \tag{3.44}$$

where λ_n = resupply arrival rate when a total of n items are in resupply. Also note that

$$\begin{aligned} \mu_\alpha^{(1)} &= \alpha\mu_1 \quad , \qquad 0 \le \alpha \le n \quad , \quad 0 \le n \le N \, , \\ \mu_{n-\alpha}^{(2)} &= (n-\alpha)\mu_2 \quad , \qquad 0 \le \alpha \le n \quad , \quad 0 \le n \le N \, , \end{aligned} \tag{3.45}$$

when $\alpha_{max} = N = \beta_{max}$ (infinite number of parallel servers in resupply at level 1 and at level 2). Observe that

$$\begin{aligned} \mu_{\alpha+1}^{(1)} P_{\alpha+1,n-\alpha}(t) &= (\alpha+1)\mu_1 f_1^{\alpha+1} f_2^{n-\alpha} \binom{n+1}{\alpha+1} P_{n+1}(t) \quad , \\ &= (n+1)\mu_1 f_1 \binom{n}{\alpha} f_1^{\alpha} f_2^{n-\alpha} P_{n+1}(t) \quad ; \end{aligned} \tag{3.46}$$

$$\therefore \qquad \sum_{\alpha=0}^{n} \mu_{\alpha+1}^{(1)} P_{\alpha+1,n-\alpha}(t) = (n+1)\mu_1 f_1 P_{n+1}(t) \quad . \tag{3.47}$$

Similarly,

$$\begin{aligned} \mu_{n-\alpha+1}^{(2)} P_{\alpha,n-\alpha+1}(t) &= (n-\alpha+1)\mu_2 f_1 f_2^{n-\alpha+1} \binom{n+1}{\alpha} \quad , \\ &= (n+1)\mu_2 f_2 \binom{n}{\alpha} f_1^{\alpha} f_2^{n-\alpha} P_{n+1}(t) \quad ; \end{aligned} \tag{3.48}$$

and

$$\sum_{\alpha=0}^{n} \mu_{n-\alpha+1}^{(2)} P_{\alpha,n-\alpha+1}(t) = (n+1)\mu_2 P_{n+1}(t) \quad ; \tag{3.49}$$

$$\begin{aligned} \mu_{\alpha}^{(1)} P_{\alpha,\beta}(t) = \alpha\mu_1 P_{\alpha,n-\alpha}(t) &= \alpha\mu_1 f_1^{\alpha} f_1^{n-\alpha} \binom{n}{\alpha} P_n(t) \quad , \\ &= \mu_1 n f_1 \binom{n-1}{\alpha-1} f_1^{\alpha-1} f_2^{n-\alpha} P_n(t) \quad ; \end{aligned} \tag{3.50}$$

$$\sum_{\alpha=0}^{n} \mu_{\alpha}^{(1)} P_{\alpha,n-\alpha}(t) = n\mu_1 f_1 P_n(t) \quad ; \tag{3.51}$$

$$\begin{aligned} \mu_{n-\alpha}^{(2)} P_{\alpha,n-\alpha}(t) &= (n-\alpha)\mu_2 \binom{n}{\alpha} f_1^{\alpha} f_2^{n-\alpha} P_n(t) \quad , \\ &= n\mu_2 f_2 \binom{n-1}{\alpha} f_1^{\alpha} f_2^{n-\alpha-1} P_n(t) \quad ; \end{aligned} \tag{3.52}$$

$$\sum_{\alpha=0}^{n} \mu_{n-\alpha}^{(2)} P_{\alpha,n-\alpha}(t) = n\mu_2 f_2 P_n(t) \quad . \tag{3.53}$$

Continuing,

$$\sum_{\alpha=0}^{n} \eta_{n-1}^{(1)} P_{\alpha-1,n-\alpha}(t) = \eta_{n-1}^{(1)} P_{n-1}(t) \quad , \tag{3.54}$$

and

$$\sum_{\alpha=0}^{n} \eta_{n-1}^{(2)} P_{\alpha,n-\alpha-1}(t) = \eta_{n-1}^{(2)} P_{n-1}(t) \tag{3.55}$$

so that

$$\sum_{\alpha=0}^{n} \left[\eta_{n-1}^{(1)} P_{\alpha-1,n-\alpha}(t) + \eta_{n-1}^{(2)} P_{\alpha,n-\alpha-1}(t)\right] = \lambda_{n-1} P_{n-1}(t) \quad . \tag{3.56}$$

Therefore, summing equation (3.40) over α with $\beta = n-\alpha$ obtains

$$\frac{\partial P_n(t)}{\partial t} = - \lambda_n P_n(t) - n\mu P_n(t) + (n+1)\mu P_{n+1}(t) + \lambda_{n-1} P_{n-1}(t) \quad ,$$

$$0 \leq n \leq N \quad , \tag{3.57}$$

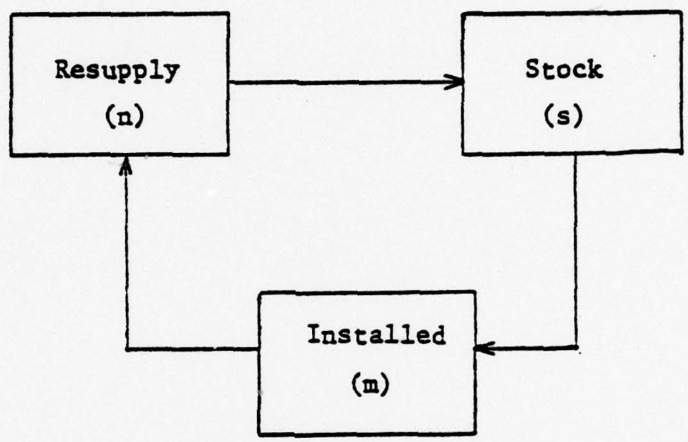
where $P_n^{(t)} = 0$ for $n < 0$ and for $n > N$.

Thus, the two index resupply state model described by equation (3.40) reduces to the one index resupply state model given by equation (3.57), and is represented by the model described in the section entitled "Conservative Parallel Server Inventory Model" and by equations (3.31) through (3.34) in this section.

SECTION 4

CONSERVATIVE PARALLEL SERVER INVENTORY MODEL

Consider the system for resupply of recoverable items that is diagrammed below.



The parameters are given as follows:

N = total number of items in the system,

n = number of items in resupply,

m = number of items installed,

s = number of items in stock,

λ = failure rate for a single item while installed,

λ_m = failure rate of installed items when number of items installed is m,

m_0 = minimum number of serviceable installed items required to accomplish mission objectives ($m \geq m_0$ implies that service performance level is being met; $m < m_0$ implies that the installed serviceable items are working at full capacity and are not fully meeting mission requirements),

m_1 = minimum number of installed serviceable items resulting in

no backorder ($m < m_1$ implies that a backorder exists, $N \geq m_1 \geq m_0 \geq 1$),

$\mu \;=\; \dfrac{1}{\tau_\mu}$ = resupply rate for a single item in resupply,

τ_μ = mean resupply time for one item in resupply,

ρ = $\lambda\tau_\mu$ = ratio of failure rate to resupply rate for one item in resupply,

μ_n = resupply rate when n items are in resupply.

For a conservative system observe that

$$N = m + s + n \quad , \tag{4.1}$$

and require that

$$\lambda_m = m_0\lambda \quad , \qquad 0 \leq n \leq N - m_0 \quad ; \tag{4.2}$$

$$\lambda_m = (N - n)\lambda \quad , \qquad N - m_0 + 1 \leq n \leq N \; ; \tag{4.3}$$

$$\mu_n = n\mu \quad , \qquad 0 \leq n \leq N \quad . \tag{4.4}$$

Thus, it is assumed that there are N parallel servers in resupply. The state transition probabilities are given by

$$\frac{\partial P_0^{(N)}(t)}{\partial t} = -m_0\lambda P_0(t) + \mu P_1(t) \quad , \qquad n = 0 \; ; \tag{4.5}$$

$$\frac{\partial P_n^{(N)}(t)}{\partial t} = -m_0\lambda P_n(t) + (n + 1)\mu P_{n+1}(t) - n\mu P_n(t) + m_0\lambda P_{n-1}(t), \tag{4.6}$$

$$0 < n \leq N - m_0 \; ;$$

$$\frac{\partial P_n^{(N)}(t)}{\partial t} = -(N-n)\lambda P_n(t) + (n+1)\mu P_{n+1}(t) - n\mu P_n(t) + (N-n+1)\lambda P_{n-1}(t), \quad (4.7)$$

$$N - m_0 + 1 \le n < N;$$

$$\frac{\partial P_N^{(N)}(t)}{\partial t} = - N\mu P_N(t) + \lambda P_{N-1}(t) , \qquad n = N . \quad (4.8)$$

In the above, $P_n^{(N)}(t)$ is the probability that n items are in resupply at time t. Using standard techniques, one obtains the following stead-state solution ($t = \infty$).

$$P_n^{(N)} = \frac{(m_0\rho)^n}{n!} P_0^{(N)} , \qquad 0 \le n \le N - m_0 ; \quad (4.9)$$

$$P_n^{(N)} = \frac{(m_0\rho)^{N-m_0} m_0!}{n!(N-n)!} \rho^{n-(N-m_0)} P_0^{(N)} , \qquad N - m_0 + 1 \le n \le N; \quad (4.10)$$

$$P_0^{(N)} = \left\{ \sum_{n=0}^{N-m_0} \frac{(m_0\rho)^n}{n!} + (m_0\rho)^{N-m_0} m_0! \sum_{n=N-m_0+1}^{N} \frac{\rho^{n-(N-m_0)}}{n!(N-n)!} \right\}^{-1} . \quad (4.11)$$

Equation (4.9) and equation (4.10) can be written

$$P_{N-x}^{(N)} = \frac{(m_0\rho)^{N-x}}{(N-x)!} P_0^{(N)} , \qquad m_0 \le x \le N , \quad (4.12)$$

$$P_{N-x}^{(N)} = (m_0\rho)^N (m_0! m_0^{-m_0}) \frac{\rho^{-x}}{(N-x)!\, x!} P_0^{(N)} , \qquad 0 \le x \le m_0 - 1 , \quad (4.13)$$

and equation (4.11) can be written

$$P_0^{(N)} = \left\{ \sum_{n=0}^{N-m_0} \frac{(m_0\rho)^n}{n!} + (m_0\rho)^N (m_0!m_0^{-m_0}) \sum_{x=0}^{m_0-1} \frac{\rho^{-x}}{x!(N-x)!} \right\}^{-1} . \tag{4.14}$$

The resupply states corresponding to $0 \leq n \leq N - m_0$ (states having full capability to fullfill mission objectives) would be described by a Poisson distribution if $P_0^{(N)}$ were given by

$$P_0^{(N)} = e^{-m_0\rho} = e^{m_0\lambda\tau_\mu} . \tag{4.15}$$

However, the Poisson process applies only if $N - m_0 \to \infty$, a condition which implies that the probability to fail to meet mission objectives is zero. Also observe that the resupply states for $N \geq n \geq N - m_0 + 1$ (those states having less than full capability to meet mission objectives) are not described by a Poisson distribution, resulting from the fact that $N - m_0$ is finite.

As observed above, the number of backorders is zero for $N \geq x \geq m_1$ ($0 \leq n \leq N - m_1$), and the number of backorders is $m_1 - x = m_1 - (N - n)$ for $0 \leq x \leq m_1 - 1$ ($N - m_1 + 1 \leq n \leq N$). Thus, the expected number of backorders $[N_B^{(N)}]$ is given by

$$[N_B^{(N)}] = \sum_{x=0}^{m_1-1} (m_1 - x) P_{N-x}^{(N)} ; \tag{4.16}$$

$$[N_B^{(N)}] = (m_0\rho)^N P_0^{(N)} \left\{ m_0!m_0^{-m_0} \sum_{x=0}^{m_0-1} (m_1-x) \frac{\rho^{-x}}{(N-x)!x!} + \sum_{x=m_0}^{m_1-1} (m_1-x) \frac{(m_0\rho)^{-x}}{(N-x)!} \right\}. \tag{4.17}$$

Inspection of equation (4.12) and equation (4.13) obtains

$$\frac{P_{N-x}^{(N)}}{P_{N-1-x}^{(N-1)}} = \frac{m_0 \rho}{N-x} \frac{P_0^{(N)}}{P_0^{(N-1)}} \quad , \qquad 0 \leq x \leq N-1; \tag{4.18}$$

$$\frac{P_n^{(N)}}{P_{n-1}^{(N-1)}} = \frac{m_0 \rho}{n} \frac{P_0^{(N)}}{P_0^{(N-1)}} \quad , \qquad 1 \leq n \leq N \quad . \tag{4.19}$$

The expected number of units in resupply is given by

$$[n^{(N)}] = \sum_{n=1}^{N} n\, P_n^{(N)} \quad , \tag{4.20}$$

so that

$$[n^{(N)}] = m_0 \rho \frac{P_0^{(N)}}{P_0^{(N-1)}} \quad , \tag{4.21}$$

or

$$\frac{P_{N-x}^{(N)}}{P_{N-1-x}^{(N-1)}} = \frac{[n^{(N)}]}{N-x} \quad . \tag{4.22}$$

Now, the probability for stockout is given by

$$p_s^{(N)} = \sum_{x=0}^{m_1 - 1} P_{N-x}^{(N)} \quad , \tag{4.23}$$

and therefore

$$[N_B^{(N)}] = [n^{(N)}]\ P_s^{(N-1)} - (N-m_1)\ P_s^{(N)} \quad . \tag{4.24}$$

Thus,

$$\frac{\partial [N_B^{(N)}]}{\partial \rho} = \frac{[n^{(N)}]}{\rho} \left\{ [N_B^{(N-1)}] - [N_B^{(N)}] \right\} \quad . \tag{4.25}$$

Define

$$\Delta [N_B^{(N)}] \equiv [N_B^{(N+1)}] - [N_B^{(N)}] \tag{4.26}$$

and

$$\Delta P_s^{(N)} \equiv P_s^{(N+1)} - P_s^{(N)} \tag{4.27}$$

to obtain

$$\frac{\partial [N_B^{(N)}]}{\partial \rho} = - \frac{[n^{(N)}]}{\rho} \Delta [N_B^{(N-1)}] \quad , \tag{4.28}$$

and

$$\frac{\partial P_s^{(N)}}{\partial \rho} = - \frac{[n^{(N)}]}{\rho} \Delta P_s^{(N-1)} \quad . \tag{4.29}$$

Also observe that equation (4.12) and equation (4.13) yield

$$\frac{1}{P_{N-x}^{(N)}} \left(\frac{\partial P_{N-x}^{(N)}}{\partial \rho} \right)_N = \frac{N-x}{\rho} + \frac{1}{P_0^{(N)}} \left(\frac{\partial P_0^{(N)}}{\partial \rho} \right)_N , \qquad 0 \leq x \leq N \ ; \tag{4.30}$$

$$\frac{1}{P_0^{(N)}} \left(\frac{\partial P_0^{(N)}}{\partial \rho} \right)_N = - \frac{[n^{(N)}]}{\rho} \quad . \tag{4.31}$$

It is observed that when $m_0 = 1$, then $\lambda_m = \lambda$ for all $m > 0$, corresponding to the situation where the failure rate is constant and independent of the number of installed serviceable units (except for $m = 0$). Thus,

$$P_n^{(N)} = \frac{\rho^n}{n!} P_0^{(N)} \quad , \qquad m_0 = 1, \quad 0 \leq n \leq N \; ; \tag{4.32}$$

$$P_0^{(N)} = \left\{ \sum_{n=0}^{N} \frac{\rho^n}{n!} \right\}^{-1} \quad , \qquad m_0 = 1 \qquad ; \tag{4.33}$$

and

$$[N_B^{(N)}] = \sum_{x=0}^{m_1 - 1} (m_1 - x) \frac{\rho^{N-x}}{(N-x)!} P_0^{(N)} \quad , \qquad m_0 = 1 \quad . \tag{4.34}$$

Note that

$$\lim_{N \to \infty} P_0^{(N)} = e^{-\rho} = e^{-\lambda \tau_\mu} \quad , \tag{4.35}$$

$$\lim_{N \to \infty} P_n^{(N)} = \frac{\rho^n}{n!} e^{-\lambda \tau_\mu} \quad , \tag{4.36}$$

and

$$\lim_{N \to \infty} [N_B^{(N)}] = \lim_{N \to \infty} \sum_{x=0}^{m_1 - 1} (m_1 - x) \frac{\rho^{N-x}}{(N-x)!} P_0^{(N)} = 0 \text{ for } m_0 = 1 \text{ and } \rho < 1 \; . \tag{4.37}$$

Thus, a Poisson demand process occurs for large N, and for large N and $\rho < 1$ the expected number of backorders is zero.

When $m_0 = m_1 = 1$, then

$$[N_B^{(N)}] = \frac{\rho^N}{N!} P_0^{(N)} = P_N^{(N)} \quad , \qquad m_0 = m_1 = 1 \; ; \tag{4.38}$$

$$[n^{(N)}] = \rho\left\{1 - \frac{\rho^N P_0^{(N)}}{N!}\right\} \quad ; \tag{4.39}$$

$$\left(\frac{\partial[N_B^{(N)}]}{\partial\rho}\right)_N = \frac{\rho^{N-1} P_0^{(N)}}{N!}\left\{N - \rho + \frac{\rho^{N+1} P_0^{(N)}}{N!}\right\} ; \tag{4.40}$$

$$[N_B^{(N+1)}] - [N_B^{(N)}] = -\frac{\rho^N}{N!} P_0^{(N)}\left\{1 - \left(\frac{\rho}{N+1}\right)\frac{P_0^{(N+1)}}{P_0^{(N)}}\right\} . \tag{4.41}$$

If $N = m_0 = m_1 = 1$, then

$$P_0^{(1)} = \frac{1}{1+\rho} , \tag{4.42}$$

$$[N_B^{(1)}] = [n^{(1)}] = \frac{\rho}{1+\rho} \quad ; \tag{4.43}$$

$$\left(\frac{\partial[N_B^{(1)}]}{\partial\rho}\right)_N = \frac{1}{(1+\rho)^2} \quad ; \tag{4.44}$$

$$[N_B^{(2)}] - [N_B^{(1)}] = -\frac{\rho(2+\rho)}{(1+\rho)(2+2\rho+\rho^2)} . \tag{4.45}$$

When $N = m_0 = m_1 = \rho = 1$, then

$$\left(\frac{\partial[N_B^{(1)}]}{\partial\rho}\right)_N = \frac{1}{4} \quad ; \tag{4.46}$$

$$[N_B^{(2)}] - [N_B^{(1)}] = -\frac{3}{10} \quad ; \tag{4.47}$$

$$[N_B^{(1)}] = \frac{1}{2} \quad . \tag{4.48}$$

A case of particular interest occurs when $N = m_0 = m_1$. This is the case when there are no extra spares and when one or more units in resupply results in system performance degradation. When $N = m_0 = m_1$, then

$$P_0^{(N)} = \frac{1}{(1+\rho)^N} \quad , \qquad m_0 = N \quad ; \tag{4.49}$$

$$P_n^{(N)} = \frac{\rho^n N!}{(1+\rho)^N n!(N-n)!} \quad , \qquad m_0 = N \quad ; \tag{4.50}$$

$$[N_B^{(N)}] = [n^{(N)}] = \frac{N\rho}{1+\rho} \quad , \qquad m_1 = m_0 = N \quad ; \tag{4.51}$$

When there is one extra spare, then

$$P_0^{(N+1)} = \frac{N+1}{1+N(1+\rho)^{N+1}} \quad , \qquad m_0 = N \quad ; \tag{4.52}$$

$$P_n^{(N+1)} = \frac{N(N+1)!}{1 + N(1+\rho)^{N+1}} \cdot \frac{\rho^n}{n!(N+1-n)!} \quad , \quad 1 \leq n \leq N+1, \; m_0 = N \; ; \tag{4.53}$$

$$[N_B^{(N+1)}] = \frac{N\{1 - (N-\rho)(1+\rho)^N\}}{1 + N(1+\rho)^N} \quad , \quad m_1 = m_0 = N \; ; \tag{4.54}$$

$$\left(\frac{\partial [N_B^{(N)}]}{\partial \rho}\right)_N = \frac{N}{(1+\rho)^2} \quad , \quad m_1 = m_0 = N \; ; \tag{4.55}$$

$$[N_B^{(N)}] - [N_B^{(N+1)}] = N \left\{ \frac{(1+\rho)^{N+1} - 1}{(1+\rho)[1 + N(1+\rho)^{N+1}]} \right\}, \quad N = m_0 = m_1 . \tag{4.56}$$

Thus,

$$\delta[N_B^{(N)}] = \frac{N}{(1+\rho)^2} \quad \text{for } N = \text{constant} \quad , \quad N = m_0 = m_1 \; ; \tag{4.57}$$

$$\delta[N_B^{(N)}] = -N \left\{ \frac{(1+\rho)^{N+1} - 1}{(1+\rho)[1 + N(1+\rho)^{N+1}]} \right\} \delta N \; , \; \delta N = 1, \text{ for } \rho = \text{constant}, \quad N = m_0 = m_1 \; . \tag{4.58}$$

SECTION 5
POISSON MODEL

In the Poisson Model the demand process described in the section entitled "Conservative Parallel Server Inventory Model" is replaced by a Poisson process as was done by Sherbrooke [104] in the development of "Metric: A Multi-Echelon Technique for Recoverable Item Control" and by Muckstadt [88, 89] in the development of "Mod-Metric: A Multi-Item, Multi-Echelon, Multi-Indenture Inventory Model." The following formulae apply when this approach is taken.

$$P_n = e^{-m_0\rho} \frac{(m_0\rho)^n}{n!} \quad , \quad 0 \le n < \infty \quad ; \tag{5.1}$$

$$[N_B^{(N)}] = \sum_{n=N-m_1+1}^{\infty} (n-N+m_1) \; P_n \quad ; \tag{5.2}$$

$$[N_B^{(N+1)}] - [N_B^{(N)}] \equiv \Delta[N_B^{(N)}] = -\mathcal{P}_s^{(N)} \quad ; \tag{5.3}$$

$$\frac{\partial [N_B^{(N)}]}{\partial \rho} = m_0 \; \mathcal{P}_s^{(N-1)} \quad ; \tag{5.4}$$

$$\frac{\partial \mathcal{P}_s^{(N)}}{\partial \rho} = m_0 P_{N-m_1+1} \quad ; \tag{5.5}$$

$$\Delta \mathcal{P}_s^{(N)} = \mathcal{P}_s^{(N+1)} - \mathcal{P}_s^{(N)} = -P_{N-m_1+1} \quad ; \tag{5.6}$$

$$[N_B^{(N)}] = [n^{(N)}] \; \mathcal{P}_s^{(N-1)} - (N-m_1)\,\mathcal{P}_s^{(N)} \quad ; \qquad (5.7)*$$

$$[n^{(N)}] = [n^{(N+1)}] = m_0\rho \quad ; \qquad (5.8)$$

$$\frac{P_{n+1}^{(N+1)}}{P_n^{(N)}} = \frac{[n^{(N+1)}]}{n+1} \quad ; \qquad (5.9)^*$$

$$\frac{P_0^{(N+1)}}{P_0^{(N)}} = 1 \quad ; \qquad (5.10)$$

$$P_0^{(N)} = e^{-m_0\rho} \quad ; \qquad (5.11)$$

$$\frac{1}{P_n^{(N)}} \frac{\partial P_n^{(N)}}{\partial \rho} = \frac{n}{\rho} - \frac{[n^{(N)}]}{\rho} \quad ; \qquad (5.12)^*$$

$$\frac{\partial [N_B^{(N)}]}{\partial \rho} = - \frac{[n^{(N)}]}{\rho} \Delta[N_B^{(N-1)}] \quad ; \qquad (5.13)^*$$

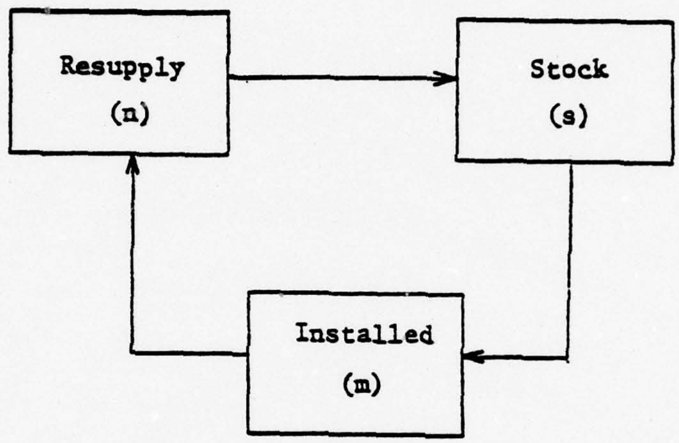
$$\frac{\partial \mathcal{P}_s^{(N)}}{\partial \rho} = - \frac{[n^{(N)}]}{\rho} \Delta \mathcal{P}_s^{(N-1)} \quad . \qquad (5.14)^*$$

Note that each of the equations whose identifying numbers are starred (*) has a form identical to that obtained for the finite population model (parallel server inventory model). The fundamental differences between the two models arise from the fact that one model assumes a finite population, while the other assumes an infinite population. In particular, note that the P_n for the Poisson Model corresponding to those states where total mission requirements cannot be fulfilled ($n > N - m_0 + 1$) have derivatives and magnitudes which are very different from those given by

the Parallel Server Model. Also, the Poisson Model defines an infinite number of states associated with the backorder condition, while the Parallel Server Model defines a finite number of states (m_1 in number).

SECTION 6
CONSERVATIVE SINGLE SERVER INVENTORY MODEL

The queueing model for recoverable items diagrammed below has been investigated, and the steady-state solution has been obtained.



The parameters for the system diagrammed above are given as follows:

N = total number of items in the system,

n = number of items in resupply,

m = number of items installed,

s = number of items in stock,

λ = failure rate for a single item while installed,

λ_m = failure rate of installed items when number of items installed is m,

m_0 = minimum number of serviceable installed items required to accomplish mission objectives ($m \geq m_0$ implies that service performance level is being met; $m < m_0$ implies that the installed serviceable items are working at full capacity and are not fully meeting mission requirements),

m_1 = minimum number of installed serviceable items resulting in

no backorder ($m < m_1$ implies that a backorder exists, $N \geq m_1 \geq m_0 \geq 1$),

$\mu \;=\; \frac{1}{\tau_\mu}$ = resupply rate for a single item in resupply,

τ_μ = mean resupply time for one item in resupply,

$\rho \;=\; \lambda\tau_\mu$ = ratio of failure rate to resupply rate for one item in resupply,

μ_n = resupply rate when n items are in resupply.

In the following development, a single server model is assumed so that

$$\mu_n \;=\; \mu \quad \text{for} \quad n \geq 1 \tag{6.1}$$

and

$$\mu_n \;=\; 0 \quad \text{for} \quad n = 0 \tag{6.2}$$

Note that when $m > m_0$, mission requirements are being met with installed serviceable items operating at less than full capacity with failure rate $\lambda_m = m_0\lambda$; when $m < m_0$, mission requirements are not being met and items are operating at full capacity with failure rate $\lambda_m = m\lambda$.

Observe that for a conservative system

$$N \;=\; m + s + n \qquad , \tag{6.3}$$

and require that

$$\lambda_m \;=\; m\lambda \quad , \quad 0 \leq m \leq m_0 \quad , \tag{6.4}$$

$$\lambda_m \;=\; m_0\lambda \quad , \quad m_0 \leq m \leq N \quad . \tag{6.5}$$

The above requirement for demand rate is equivalent to

$$\lambda_m = (N - n)\lambda \quad , \quad N - m_0 + 1 \leq n \leq N \, , \tag{6.6}$$

$$\lambda_m = m_0\lambda \quad , \quad 0 \leq n \leq N - m_0 \quad . \tag{6.7}$$

The state transition probabilities are given by

$$\left(\frac{\partial P_0^{(N)}(t)}{\partial t}\right) = -m_0\lambda P_0^{(N)}(t) + \mu P_1^{(N)}(t) \tag{6.8}$$

$$\left(\frac{\partial P_n^{(N)}(t)}{\partial t}\right) = -m_0\lambda P_n^{(N)}(t) + \mu P_{n+1}^{(N)}(t) - \mu P_n^{(N)}(t) + m_0\lambda P_{n-1}^{(N)}(t),$$

$$0 < n \leq N - m_0 \; ; \tag{6.9}$$

$$\left(\frac{\partial P_n^{(N)}(t)}{\partial t}\right) = -(N-n)\lambda P_n^{(N)}(t) + \mu P_{n+1}^{(N)}(t) - \mu P_n^{(N)}(t) + (N-n+1)\lambda P_{n-1}^{(N)}(t),$$

$$N > n \geq N - m_0 + 1 \; ; \tag{6.10}$$

$$\left(\frac{\partial P_N^{(N)}(t)}{\partial t}\right) = - \mu P_N^{(N)}(t) + \lambda P_{N-1}^{(N)}(t) \quad ,$$

$$n = N \, . \tag{6.11}$$

In the above, $P_n^{(N)}(t)$ is the probability that n items are in resupply at time t. Using standard techniques, one obtains the following steady-state solution $(t = \infty)$.

$$P_n^{(N)} = (m_0\rho)^n P_0^{(N)} \quad , \qquad 0 \le n \le N - m_0 \quad , \tag{6.12}$$

$$P_n^{(N)} = (m_0\rho)^{N-m_0} \frac{m_0!}{(N-n)!} \rho^{n-(N-m_0)} P_0^{(N)} \quad , \quad N - m_0 + 1 \le n \le N \; . \tag{6.13}$$

Equation (6.13) can be rewritten

$$P_{N-x}^{(N)} = (m_0\rho)^N m_0^{-m_0} m_0! \frac{\rho^{-x}}{x!} P_0^{(N)} \quad , \qquad 0 \le x \le m_0 - 1 \quad , \tag{6.14}$$

where $x \equiv N - n$. The probability that there are zero units in resupply is

$$P_0^{(N)} = \frac{1 - m_0\rho}{1 - (m_0\rho)^{N-m_0+1} + (1 - m_0\rho)(m_0\rho)^N m_0^{-m_0} m_0! G_{m_0}(\rho)} \quad , \tag{6.15}$$

where

$$G_{m_0}(\rho) \equiv \sum_{x=0}^{m_0-1} \frac{\rho^{-x}}{x!} \quad . \tag{6.16}$$

Note that the backorder level is zero for $0 \le n \le N - m_1$ and is equal to $m_1 - x = m_1 - (N - n)$ for $N - m_1 + 1 \le n \le N$. Thus, the expected backorder level is

$$[N_B^{(N)}] = \sum_{x=1}^{m_1-1} (m_1 - x) P_{N-x}^{(N)} \tag{6.17}$$

so that

$$[N_B^{(N)}] = \sum_{x=0}^{m_0-1} (m_1 - x)(m_0\rho)^N (m_0! m_0^{-m_0}) P_0^{(N)} \frac{\rho^{-x}}{x!} + \sum_{x=m_0}^{m_1-1} (m_1 - x)(m_0\rho)^{N-x} P_0^{(N)} . \quad (6.18)$$

Solving for $N = N([N_B^{(N)}], \rho)$ obtains

$$\left\{-\ell n(m_0\rho)\right\} \left\{N(\rho, [N_B^{(N)}])\right\} =$$

$$-\ell n[N_B^{(N)}] + \ell n\left\{ (1-m_0\rho)\left[\sum_{x=0}^{m_0-1} (m_1-x) m_0!\, m_0^{-m_0} \frac{\rho^{-x}}{x!} + \sum_{x=0}^{m_1-1} (m_1-x)(m_0\rho)^{-x}\right]\right.$$

$$\left. + [N_B^{(N)}]\left[(m_0\rho)^{-m_0+1} - (1 - m_0\rho) m_0^{-m_0} m_0!\, G_{m_0}(\rho)\right]\right\} . \quad (6.19)$$

If $m_0 = m_1$, then

$$[N_B^{(N)}] = \frac{(m_0\rho)^N m_0^{-m_0} m_0!(1 - m_0\rho) \left\{ m_0 G_{m_0}(\rho) - \frac{1}{\rho} G_{m_0-1}(\rho)\right\}}{1 - (m_0\rho)^{N-m_0+1} + (1 - m_0\rho)(m_0\rho)^N m_0^{-m_0} m_0! G_m(\rho)}, \quad m_0 = m_1, \quad (6.20)$$

and

$$\left\{\ell n(m_0\rho)\right\} \left\{N(\rho, [N_B^{(N)}]\right\} = \ell n[N_B^{(N)}] - \ell n\left\{ m_0^{-m_0} m_0!(1 - m_0\rho) G_{m_0}(\rho)\{m - [N_B^{(N)}]\}\right.$$

$$+ [N_B^{(N)}](m_0\rho)^{-m_0+1} \quad (6.21)$$

$$\left. - \frac{1}{\rho} m_0^{-m_0} m_0!(1-m_0\rho) G_{m_0-1}(\rho)\right\},$$

$$m_0 = m_1 .$$

For a given fixed value of $[N_B^{(N)}]$, equation (6.19) gives the relationship of N and ρ. Thus, all values of (N, ρ) that yield a particular value of $[N_B^{(N)}]$ may be obtained for comparison of inventory investment level to resupply time level for purposes of trade-off analysis described earlier in this document.

Note that when $m_0 = 1$, then $\lambda_m = \lambda$ for all $m > 0$, corresponding to the situation where the failure rate is constant and independent of the number of installed serviceable units (except for $m = 0$). Returning to equation (6.15) and equation (6.18), one obtains

$$P_n^{(N)} = \frac{\rho^n (1 - \rho)}{1 - \rho^{N+1}} \quad , \quad m_0 = 1 \quad , \qquad 0 \leq n \leq N \quad ; \tag{6.22}$$

$$P_0^{(N)} = \frac{1 - \rho}{1 - \rho^{N+1}} \quad , \qquad m_0 = 1 \quad ; \tag{6.23}$$

and

$$[N_B^{(N)}] = \frac{\rho^{N-m_1+1}}{(1 - \rho)(1 - \rho^{N+1})} \left\{ 1 - \rho^{m_1}[m_1(1 - \rho) + 1] \right\} , \quad m_0 = 1 \, . \tag{6.24}$$

Further,

$$N(\rho, [N_B^{(N)}]) = \frac{(m_1 - 1)\ln\rho + \ln[N_B^{(N)}] + \ln(1-\rho) - \ln\left\{ 1 - \rho^{m_1}\left[1 + (1-\rho)(m_1 - [N_B^{(N)}])\right]\right\}}{\ln\rho} , \tag{6.25}$$

$$m_0 = 1 \, .$$

Thus, when $m_0 = m_1 = 1$

$$[N_B^{(N)}] = \frac{(1-\rho)\,\rho^N}{1-\rho^{N+1}} \quad , \qquad m_0 = m_1 = 1; \qquad (6.26)$$

$$N(\rho,[N_B^{(N)}]) = \frac{\ell n[N_B^{(N)}] - \ell n\left\{1 - \rho(1 - [N_B^{(N)}])\right\}}{\ell n\rho}, \qquad m_0 = m_1 = 1; \qquad (6.27)$$

$$\left(\frac{\partial N}{\partial \rho}\right)_{[N_B^{(N)}]} = \frac{N/\rho - [(1-\rho^N)/(1-\rho)]}{(-\ell n\rho)} \quad , \qquad m_0 = m_1 = 1; \qquad (6.28)$$

$$\left(\frac{\partial [N_B^{(N)}]}{\partial N}\right)_\rho = \frac{\ell n\rho}{1-\rho^{N+1}} \cdot \frac{(1-\rho)\rho^N}{1-\rho^{N+1}} \quad , \qquad m_0 = m_1 = 1; \qquad (6.29)$$

$$\left(\frac{\partial [N_B^{(N)}]}{\partial \rho}\right)_N = \frac{\frac{N}{\rho} - \frac{(1-\rho^N)}{1-\rho}}{1-\rho^{N+1}} \cdot \frac{(1-\rho)\rho^N}{1-\rho^{N+1}} \quad , \qquad m_0 = m_1 = 1. \qquad (6.30)$$

Now consider $\rho = m_1 = m_0 = 1$. Then

$$[N_B^{(N)}] = \frac{1}{N+1} \; ; \left(\frac{\partial [N_B^{(N)}]}{\partial N}\right)_\rho = -\frac{1}{(N+1)^2} \; ; \quad \left(\frac{\partial [N_B^{(N)}]}{\partial \rho}\right)_N = \frac{N}{2(N+1)} \; ;$$

$$\left(\frac{\partial N}{\partial \rho}\right)_{[N_B^{(N)}]} = \frac{N(N+1)}{2} \quad ;$$

and

$$\delta N = \frac{N(N+1)}{2}\delta\rho \text{ for } [N_B^{(N)}] = \text{constant} \quad . \tag{6.31}$$

When $N = \rho = m_1 = m_0 = 1$, then

$$[N_B^{(1)}] = \frac{1}{2} \; ; \; \left(\frac{\partial[N_B^{(1)}]}{\partial N}\right)_\rho = -\frac{1}{4} \; ; \; \left(\frac{\partial[N_B^{(1)}]}{\partial\rho}\right)_N = \frac{1}{4} \; ;$$

$$\left(\frac{\partial N}{\partial\rho}\right)_{[N_N^{(1)}]} = 1 \quad .$$

Thus, if $[N_B^{(1)}]$ = constant, then

$$\delta N = \lambda\delta\tau_\mu \quad ; \tag{6.32}$$

if ρ = constant, then

$$\delta[N_B^{(1)}] = -\frac{1}{4}\delta N \quad ; \tag{6.33}$$

if N = constant, then

$$\delta[N_B^{(1)}] = \frac{1}{4}\lambda\delta\tau_\mu \quad . \tag{6.34}$$

It is interesting to calculate $[N_B^{(N+1)}] - [N_B^{(N)}]$ and $\left(\frac{\partial[N_B^{(N)}]}{\partial\rho}\right)_N$ in a fashion similar to that used in the case of the conservative parallel server model. Inspection of equation (7.12) and equation (7.13) obtains

$$P_{N-x+1}^{(N+1)} = (m_0\rho)\,\frac{P_0^{(N+1)}}{P_0^{(N)}}\,P_{N-x}^{(N)} \quad , \qquad 0 \le x \le N \; ; \qquad (6.35)$$

and returning to equation (7.17) leads to

$$\frac{[N_B^{(N)}] - [N_B^{(N+1)}]}{[N_B^{(N)}]} = 1 - m_0\rho\,\frac{P_0^{(N+1)}}{P_0^{(N)}} > 0 \quad . \qquad (6.36)$$

Equation (6.12) and equation (6.13) yield

$$\frac{1}{P_{N-x}^{(N)}}\left(\frac{\partial P_{N-x}^{(N)}}{\partial\rho}\right)_N = \frac{N-x}{\rho} + \frac{1}{P_0^{(N)}}\left(\frac{\partial P_0^{(N)}}{\partial\rho}\right)_N \quad , \qquad 0 \le x \le N \; ; \quad (6.37)$$

$$\frac{1}{P_0^{(N)}}\left(\frac{\partial P_0^{(N)}}{\partial\rho}\right)_N = -\frac{[n^{(N)}]}{\rho} \quad . \qquad (6.38)$$

Thus,

$$\left(\frac{\partial[N_B{}^{(N)}]}{\partial\rho}\right)_N = \frac{[N_B{}^{(N)}]}{\rho}\left\{(N-m_1) - [n^{(N)}] + \frac{[N_B{}^{(N)}]^2}{[N_B{}^{(N)}]}\right\} > 0 \quad , \qquad (6.39)$$

an expression which is identical in form to that obtained for the parallel server case.

Other formulae which apply are given below.

$$[N_B{}^{(N+1)}] - [N_B{}^{(N)}] \equiv \Delta[N_B{}^{(N)}] = -P_o^{(N+1)}\,[N_B{}^{(N)}] \quad ; \; (6.40)$$

$$p_s^{(N+1)} - p_s^{(N)} \equiv \Delta p_s^{(N)} = -P_0^{(N+1)} \; p_s^{(N)} \quad ; \tag{6.41}$$

$$\frac{P_{n+1}^{(N)}}{P_n^{(N)}} = m_0 \rho \quad , \quad 0 \leq n \leq N - m_0 \quad ;$$

$$\frac{P_{N-x+1}^{(N)}}{P_{N-x}^{(N)}} = m_0 \rho \quad , \quad N - 1 \geq x \geq m_0 \quad ; \tag{6.42}$$

$$\frac{P_{n+1}^{(N)}}{P_n^{(N)}} = \rho(N-n) \quad , \quad N - m_0 + 1 \leq n \leq N - 1 \quad ;$$

$$\frac{P_{N+1-x}^{(N)}}{P_{N-x}^{(N)}} = \rho x \quad , \quad 1 \leq x \leq m_0 - 1 \quad ; \tag{6.43}$$

$$m_0 \rho \; \frac{P_0^{(N+1)}}{P_0^{(N)}} = 1 - P_0^{(N+1)} \quad . \tag{6.44}$$

SECTION 7
EXAMPLE COST FUNCTION

The total system cost is given by

$$C = C(N, \rho) \quad , \tag{7.1}$$

where

N = number of inventory investment items,

$\rho = \dfrac{\lambda}{\mu} = \dfrac{\tau_\mu}{\tau_\lambda}$ = ratio of mean resupply time to mean time between demands.

Total system cost is expected to increase linearly with increasing N and to increase with decreasing ρ (decreasing τ_μ). For purposes of exposition, assume a simple cost function of the form

$$C(N, \rho) = c_1 N + \frac{c_2 N}{\rho} \quad , \tag{7.2}$$

where

c_1 = cost per investment item,

c_2 = cost per investment item per unit ρ.

Thus,

inventory investment cost = $c_1 N$,

cost for resupply system = $\dfrac{c_2 N}{\rho}$.

Define

$$z(N, \rho) \equiv \frac{C(N, \rho)}{c_1} = N\left(1 + \frac{c_2}{c_1 \rho}\right) \quad , \tag{7.3}$$

and

$$\rho_0 \equiv \frac{c_2}{c_1} \quad . \tag{7.4}$$

Thus,

$$z(N, \rho) = N \left(1 + \frac{\rho_0}{\rho}\right) \quad . \qquad (7.5)$$

$z(N, \rho)$ is a unitless cost function expressed in terms of number of equivalent investment items.

$$z = z_1 + z_2 \quad , \qquad (7.6)$$

$z_1 = N =$ number of investment items ,

$z_2 = \dfrac{N\rho_0}{\rho} =$ cost of resupply system in terms of equivalent investment items .

If $\dfrac{\rho_0}{\rho} = 1$, then for every dollar spent for inventory investment an equal amount is spent for the resupply system.

SECTION 8
PROBLEM FORMULATION

The mathematical statement of the problem takes on two possible forms.

PROBLEM 1

Minimize

$$z = z(N, \rho) \tag{8.1}$$

Subject to

$$\begin{aligned} [N_B(N, \rho)] &= N_B^{(0)} = \text{constant} \quad , \\ N &= \text{integer} > 0 \\ \rho &> 0 \quad . \end{aligned} \tag{8.2}$$

PROBLEM 2

Minimize

$$[N_B] = [N_B(N, \rho)] \tag{8.3}$$

Subject to

$$\begin{aligned} z(N, \rho) &= z_0 = \text{constant} \quad , \\ N &= \text{integer} > 0 \quad , \\ \rho &> 0 \quad . \end{aligned} \tag{8.4}$$

SECTION 9
PROBLEM SOLUTION

In this section are presented the procedures to be followed for problem solution. PROBLEM 1 and PROBLEM 2 are restated below, and associated optimality conditions are given.

PROBLEM 1

Minimize

$$z = z(N, \rho) \quad , \tag{9.1}$$

Subject to

$$\rho = \rho(N_B^{(0)}, N) = \rho(N), \text{ with } N_B^{(0)} \text{ fixed.} \tag{9.2}$$

$$z = N\left(1 + \frac{\rho_0}{\rho(N_B^{(0)}, N)}\right) \quad , \tag{9.3}$$

$$z = N\left(1 + \frac{\rho_0}{\rho(N)}\right) \quad . \tag{9.4}$$

Since N is an integer,

$$z\{N+1, \rho(N+1)\} - z\{N, \rho(N)\} \geq 0 \tag{9.5}$$

and

$$z\{N-1, \rho(N-1)\} - z\{N, \rho(N)\} \geq 0 \quad , \tag{9.6}$$

when z is a minimum.

PROBLEM 2

Minimize

$$[N_B] = [N_B(N, \rho)] \quad , \tag{9.7}$$

Subject to

$$\rho = \rho(N, z_0) = \rho(N) \quad , \quad \text{with } z_0 \text{ fixed.} \tag{9.8}$$

$$\rho(N) = \frac{N\rho_0}{z_0 - N} \quad , \quad N < z_0 \quad ; \tag{9.9}$$

$$[N_B(N, \rho)] = [N_B\{N, \rho(N)\}] \quad . \tag{9.10}$$

If N is an integer, then

$$[N_B\{N+1, \rho(N+1)\}] \geq [N_B\{N, \rho(N)\}] \tag{9.11}$$

and

$$[N_B\{N-1, \rho(N-1)\}] \geq [N_B\{N, \rho(N)\}] \quad , \tag{9.12}$$

when $[N_B]$ is a minimum.

Both of the above integer programming problems are solved by a simple search utilizing a gradient technique (described in Section 10) that obtains solutions in two to six interactions. A similar technique is utilized to obtain solutions to the equation

$$\rho = \rho(N_B^{(0)}, N) \quad ,$$

where ρ is to be determined when N and the backorder level $N_B^{(0)}$ are

specified. In general, PROBLEM 1 requires more computing time than that required for PROBLEM 2. The solution to $\rho = \rho(N_B^{(o)}, N)$ requires a great deal more time to obtain than that required for the solution of

$$\rho(N) = \frac{N\rho_0}{z_0 - N} \quad .$$

The mathematical programming problems described above have been thoroughly investigated for the backorder function described by (1) the Parallel Server Model and for (2) the Poisson Model. Comparisons and conclusions are given in Section 12.

SECTION 10

A GRADIENT TECHNIQUE TO SOLVE A SET OF COUPLED EQUATIONS

Let

$$\underline{r} \equiv \begin{pmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ \cdot \\ x_n \end{pmatrix} \quad \text{and} \quad \underline{\epsilon} \equiv \begin{pmatrix} \epsilon_1 \\ \epsilon_2 \\ \cdot \\ \cdot \\ \cdot \\ \epsilon_n \end{pmatrix} \tag{10.1}$$

be n component column vectors and the gradient operator $\underline{\nabla}$ be defined to be the row vector

$$\underline{\nabla} \equiv \left(\frac{\partial}{\partial x_1} , \frac{\partial}{\partial x_2} , \ldots , \frac{\partial}{\partial x_n} \right) . \tag{10.2}$$

It is desired to find a solution to the set of n simultaneous equations

$$g_i(\underline{r}) = 0 , \quad i = 1, 2, 3, \ldots, n \tag{10.3}$$

in n unknowns represented by the column vector $\underline{r}$. The first order Taylor expansion about a given point $\underline{r}_o$ is

$$g_i(\underline{r}_o + \underline{\epsilon}) = \underline{\nabla} g_i \underline{\epsilon} + g_i(\underline{r}_o) , \tag{10.4}$$

where $\underline{\nabla} g_i = \underline{\nabla} g_i(\underline{r}_o)$; i.e., the first derivatives are evaluated at the point $\underline{r}_o$, and $\underline{r}_o$ is assumed to be an approximate solution to equations (10.3). If $\underline{r} = \underline{r}_o + \underline{\epsilon}$ is to be a solution to equation (10.3), then

$$g_i(\underline{r}_o + \underline{\epsilon}) = 0, \quad i = 1, 2, \ldots, n \tag{10.5}$$

and

$$\underline{\nabla} g_i \underline{\epsilon} = -g_i(\underline{r}_o) \quad , \quad i = 1, 2, \ldots, n \quad . \tag{10.6}$$

Restating equation (10.6) leads to

$$\sum_{j=1}^{n} (\underline{\nabla} g_i)_j \quad \epsilon_j = -g_i(\underline{r}_o) \quad . \tag{10.7}$$

Now define the square n x n matrix $\underline{G}$ by

$$\underline{G} \equiv \begin{pmatrix} \left(\frac{\partial g_1}{\partial x_1}\right)_0 & \left(\frac{\partial g_1}{\partial x_2}\right)_0 & \cdots & \left(\frac{\partial g_1}{\partial x_n}\right)_0 \\ \left(\frac{\partial g_2}{\partial x_1}\right)_0 & \left(\frac{\partial g_2}{\partial x_2}\right)_0 & \cdots & \left(\frac{\partial g_2}{\partial x_n}\right)_0 \\ \cdot & & & \\ \cdot & & & \\ \cdot & & & \\ \left(\frac{\partial g_n}{\partial x_1}\right)_0 & \left(\frac{\partial g_n}{\partial x_2}\right)_0 & \cdots & \left(\frac{\partial g_n}{\partial x_n}\right)_0 \end{pmatrix} = \begin{pmatrix} \underline{\nabla} g_1(\underline{r}_o) \\ \\ \underline{\nabla} g_2(\underline{r}_o) \\ \\ \\ \\ \underline{\nabla} g_n(\underline{r}_o) \end{pmatrix} \tag{10.8}$$

and the n component column vector $g(\underline{r}_o)$ by

$$\underline{g}(\underline{r}_o) = \begin{pmatrix} g_1(\underline{r}_o) \\ g_2(\underline{r}_o) \\ \cdot \\ \cdot \\ \cdot \\ g_n(\underline{r}_o) \end{pmatrix} \quad . \tag{10.9}$$

In the above it is understood that all derivatives $\left(\frac{\partial g_i}{\partial x_j}\right)_0$ and all functions $g_i(\underline{r}_o)$ are evaluated at the point $\underline{r} = \underline{r}_o$. It is observed that

$$[\underline{G}(\underline{r}_o)]_{ij} = [\nabla g_i(\underline{r}_o)]_j \quad \text{and} \quad [\underline{g}(\underline{r}_o)]_i = g_i(\underline{r}_o) \quad . \tag{10.10}$$

Therefore,

$$\sum_{j=1}^{n} G_{ij}(\underline{r}_o)\epsilon_j = -g_i(\underline{r}_o) \quad ;$$

$$\underline{G}(\underline{r}_o)\underline{\epsilon} = -\underline{g}(\underline{r}_o) \quad . \tag{10.11}$$

The above implies that

$$\underline{\epsilon} = -\underline{G}^{-1}(\underline{r}_o)\underline{g}(\underline{r}_o) \quad , \tag{10.12}$$

or

$$\underline{\epsilon} = -\underline{G}^{-1}\underline{g} \tag{10.13}$$

if the matrix $\underline{G}^{-1}$ exists (i.e., if $\det|\underline{G}| \neq 0$). The individual components of $\underline{\epsilon}$ are given by

$$\epsilon_j = -\sum_{k=1}^{n} G^{-1}_{jk} g_k \quad , \quad j = 1, 2, \ldots, n. \tag{10.14}$$

As long as $\underline{r}_o$ is "reasonably close" to the particular solution desired and no pathologic conditions exist (forcing $\det|\underline{G}| = 0$), then equation (10.13) can be used to successive arrive at values of $\underline{r}$ which satisfy equation (10.3) as precisely as desired.

The procedure simply stated is as follows:

(i) select an initial trial solution value $\underline{r} = \underline{r}_i$, where i = the iteration number (i = 0 initially);

(ii) apply equation (10.13) to determine the correction vector $\underline{\epsilon}_i$;

(iii) check to see if $\underline{\epsilon}_i$ satisfies $|\underline{\epsilon}_i| < \underline{\epsilon}_{max}$, where the $(\underline{\epsilon}_{max})_j = (\epsilon_j)_{max} > 0$ represent upper limit values on the errors to be tolerated in the solution values for each of the variables x_j;

(iv) (a) if $|\underline{\epsilon}_i| > \underline{\epsilon}_{max}$, then set $\underline{r}_{i+1} = \underline{r}_i + \underline{\epsilon}_i$ (i.e., calculate a new trial solution to the system of equations $\underline{g}(\underline{r}) = \underline{0}$) and return to step (ii) above with i = i + 1; (b) if $|\underline{\epsilon}_i| < \underline{\epsilon}_{max}$, then stop and calculate $\underline{r}_{i+1} = \underline{r}_i + \underline{\epsilon}_i$, where $\underline{r}_{i+1}$ represents the desired solution to $\underline{g}(\underline{r}) = \underline{0}$.

As an example, consider the problem

$$g_1(x, y) = x^2 - y^2 - 4 = 0 \quad ,$$
$$g_2(x, y) = x + y - 1 = 0 \quad .$$

The solution to the above problem is readily obtained by conventional techniques:

$$x + y = 1 \quad ;$$
$$x - y = 4 \quad ;$$
$$2x = 5$$
$$x = 5/2 \quad \text{and} \quad y = -3/2 \quad .$$

To illustrate the technique developed above proceed as follows:

$$\underline{\nabla} g_1 = (2x, -2y) \quad ,$$
$$\underline{\nabla} g_2 = (1, 1) \quad ; \qquad \underline{G} = \begin{pmatrix} 2x & -2y \\ 1 & 1 \end{pmatrix} \quad ;$$
$$\det|\underline{G}| = 2(x + y) \quad ;$$
$$\underline{G}^{-1} = \frac{1}{2(x+y)} \begin{pmatrix} 1 & 2y \\ -1 & 2x \end{pmatrix} \quad ; \quad g = \begin{pmatrix} x^2 - y^2 - 4 \\ x + y - 1 \end{pmatrix} \quad .$$

Choose $\underline{r}_o = \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$ as the initial trial solution.

$$\underline{G}^{-1}(\underline{r}_o) = \frac{1}{4}\begin{pmatrix} 1 & 2 \\ -1 & 2 \end{pmatrix} \quad ; \quad \underline{g}(\underline{r}_o) = \begin{pmatrix} -4 \\ +1 \end{pmatrix} \quad ;$$

$$\underline{\epsilon}_o = -\begin{pmatrix} \frac{1}{4} & \frac{1}{2} \\ -\frac{1}{4} & \frac{1}{2} \end{pmatrix}\begin{pmatrix} -4 \\ 1 \end{pmatrix} = \begin{pmatrix} \frac{1}{2} \\ -\frac{3}{2} \end{pmatrix} \neq \underline{0} \quad .$$

Iteration No. 1 proceeds as follows:

$$\underline{r}_1 = \underline{r}_o + \underline{\epsilon}_o = \begin{pmatrix} \frac{3}{2} \\ -\frac{1}{2} \end{pmatrix}$$

$$\underline{G}^{-1}(\underline{r}_1) = \frac{1}{2}\begin{pmatrix} 1 & -1 \\ -1 & 3 \end{pmatrix} = \begin{pmatrix} \frac{1}{2} & -\frac{1}{2} \\ -\frac{1}{2} & \frac{3}{2} \end{pmatrix} \quad ; \quad \underline{g}(\underline{r}_1) = \begin{pmatrix} -2 \\ 0 \end{pmatrix} \quad ;$$

$$\underline{\epsilon}_1 = -\begin{pmatrix} \frac{1}{2} & -\frac{1}{2} \\ -\frac{1}{2} & \frac{3}{2} \end{pmatrix}\begin{pmatrix} -2 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ -1 \end{pmatrix} \neq \underline{0} \quad .$$

Iteration No. 2 proceeds as follows:

$$\underline{r}_2 = \underline{r}_1 + \underline{\epsilon}_1 = \begin{pmatrix} \frac{3}{2} \\ -\frac{1}{2} \end{pmatrix} + \begin{pmatrix} 1 \\ -1 \end{pmatrix} = \begin{pmatrix} \frac{5}{2} \\ -\frac{3}{2} \end{pmatrix}$$

$$\underline{G}^{-1}(\underline{r}_2) = \frac{1}{2}\begin{pmatrix} 1 & -3 \\ -1 & 5 \end{pmatrix} = \begin{pmatrix} \frac{1}{2} & -\frac{3}{2} \\ -\frac{1}{2} & \frac{5}{2} \end{pmatrix} \quad ; \quad \underline{g}(\underline{r}_2) = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

$$\therefore\ \underline{\epsilon}_3 = 0 \quad \text{and} \quad \underline{r}_3 = \begin{pmatrix} \frac{5}{2} \\ \\ -\frac{3}{2} \end{pmatrix} \text{ is a solution.}$$

Note that it is not necessary that the first approximation $\underline{r}_o$ be terribly good; what is required is that the $\underline{r}_o$ chosen be "closer" to the solution of interest than to any other possible solution (thus preventing the possibility of moving along in the wrong direction) and that in successive calculations no pathological conditions causing det$|\underline{G}|$ to be zero occur [such as encountering a saddle point or a relative minimum or maximum for any of the $g_i(\underline{x})$]. If such a condition does arise, however, it is still possible to apply the technique by "skipping over" the critical point encountered [Note that det$|\underline{G}|$ = 0 for any $\underline{x}$ which is a critical point for any of the $g_i(\underline{x})$ in $g(\underline{x})$.]. A "good" value for $\underline{r}_o$ eliminates such a possibility from occurring, and in general no such problem occurs in practical applications.

The one dimensional case of the solution technique is a special case of the development presented above. To obtain a solution to the equation

$$g(x) = 0 \tag{10.15}$$

note that

$$g(x_o + \epsilon) = \left(\frac{\partial g(x)}{\partial x}\right)_0 \epsilon + g(x_o) \tag{10.16}$$

and

$$\epsilon = -g(x_o)\left(\frac{\partial g}{\partial x}\right)_0^{-1} \tag{10.17}$$

if $x = x_o + \epsilon$ is to be a solution to equation (10.15). The procedure to determine the solution is precisely the same as that given in steps (i) through (iv) above.

An example illustration of the one dimensional case is given below. In order to solve

$$x + \ln x = 0$$

for x, one proceeds as follows:

$$g(x) = x + \ln x \quad ;$$

$$\frac{\partial g}{\partial x} = 1 + \frac{1}{x} \quad ;$$

$$\epsilon = -\frac{x(x + \ln x)}{1 + x} \quad .$$

Try x = 1 as a trial solution. Then

$$\epsilon_o = -0.500.$$

Iteration No. 1 proceeds as follows:

$$x_1 = x_o + \epsilon_o = 0.5 \quad ;$$

$$\epsilon_1 = \frac{-0.5(0.5 - 0.693147)}{1.5} = +0.0643824 \quad .$$

Iteration No. 2 proceeds as follows:

$$x_2 = x_1 + \epsilon_1 = 0.5 + 0.0643824 = 0.5643824 \quad ;$$

$$\epsilon_2 = \frac{-(0.5643824)(0.5643824 - 0.5720233)}{1.5643824} \quad ;$$

$$\epsilon_2 = +0.00275659 \quad .$$

Iteration No. 3 proceeds as follows:

$$x_3 = x_2 + \epsilon_2 = 0.5671389835 \quad ;$$

$$\epsilon_3 = \frac{-(0.5671389835)(0.5671389835 - 0.5671508844)}{1.5671390} \quad ;$$

$$\epsilon_3 = +0.0000043069 \quad .$$

Iteration No. 4 proceeds as follows:

$$x_4 = x_3 + \epsilon_3 = 0.5671432904 \quad ;$$

$$\epsilon_4 = \frac{-(0.5671432904)(0.5671432904 - 0.5671432904)}{1.5671432904} \quad ;$$

$$\epsilon_4 = 0 \quad .$$

$$\therefore \quad x = 0.5671432904 \text{ to nine significant figures} \quad .$$

If a solution value to the equation $g(x) = 0$ is desired when x is constrained to be an integer, then the above procedure is modified as follows.

$$g(x_o + \epsilon) = g(x_o) + \Delta g(x_o)\epsilon \cong 0 \quad , \tag{10.18}$$

$$\Delta g(x_o) \equiv g(x_o + 1) - g(x_o) \quad , \tag{10.19}$$

$$\epsilon = \left[- \frac{g(x_o)}{\Delta g(x_o)} \right]_T \quad , \tag{10.20}$$

where the subscript T means set ϵ = integer = truncated value of the numerical quantity enclosed in the brackets. The iterative procedure is then that given by steps (i) through (iv) given earlier in this section with equation (10.20) being substituted for equation (10.17) and the initial value for x_o being set equal to an integer. If no solution to $g(x) = 0$ exists (this is possible since x = integer), then the above procedure may be utilized in finding the value of $x \geq 0$ that minimizes $|g(x)|$ and the value of $x < 0$ (if allowed) that minimizes $|g(x)|$.

In the solution procedure for PROBLEM 1, the gradient technique is applied to determine ρ in equation (9.2) and is used to aid in the search for the optimal value of N which satisfies the conditions expressed in relations (9.5) and (9.6). The search for (N, ρ) is conducted along a curve of constant $[N_B]$. In the solution procedure for PROBLEM 2, the gradient technique is

used to aid in the search for the optimal value of N which satisfies the conditions expressed in relations (9.11) and (9.12). Here, the search for (N, ρ) is conducted along a curve of constant z.

The solution procedure developed here for PROBLEM 1 and PROBLEM 2 is easily adapted to any cost function of the form $C = C(N, \rho)$ and for any backorder function of the form $[N_B] = [N_B(N, \rho)]$. In this study, PROBLEM 1 and PROBLEM 2 have been investigated with the backorder functions described by the Poisson Model (Section 5) and the Parallel Server Model (Section 4) and with $z = z(N, \rho)$ as described in Section 7.

SECTION 11
SENSITIVITY ANALYSIS

It is of interest to determine the range of ρ for which the optimum value of N remains unchanged. For PROBLEM 1, application of the optimality condition and the corresponding constraint yields values for $(\rho_0)_{max}$ and $(\rho_0)_{min}$ such that when

$$(\rho_0)_{min} \leq \rho_0 \leq (\rho_0)_{max} \tag{11.1}$$

the current optimal value obtained for the (N, ρ) ordered pair remains unchanged; i.e., the optimal solution to PROBLEM 1 is independent of the choice of ρ_0 within the interval $(\rho_0)_{min} \leq \rho_0 \leq (\rho_0)_{max}$. Similarly, application of the optimality condition for PROBLEM 2 and the corresponding constraint yields values for $(\rho_0)_{max}$ and $(\rho_0)_{min}$ such that when

$$(\rho_0)_{min} \leq \rho_0 \leq (\rho_0)_{max} \tag{11.2}$$

the current optimal value of N remains unchanged; i.e., the optimal solution value for N for PROBLEM 2 is independent of the choice of ρ_0 within the interval $(\rho_0)_{max} \geq \rho_0 \geq (\rho_0)_{min}$. However, in the case for PROBLEM 2, if ρ_0 is allowed to vary and the optimum value of N remains fixed, then the optimum value of ρ must change to keep z = constant.

Let (N_1, ρ_1) be an optimal solution to PROBLEM 1. Define $\Delta\rho_1 \geq 0$ and $\Delta\rho_2 \geq 0$ by

$$[N_B(N_1 + 1, \rho_1 + \Delta\rho_1)] = N_B^{(0)} , \tag{11.3}$$

$$[N_B(N_1 - 1, \rho_1 - \Delta\rho_2)] = N_B^{(0)} . \tag{11.4}$$

Condition (9.5) in Section 9 becomes

$$(N_1 + 1) \left(1 + \frac{\rho_0}{\rho_1 + \Delta\rho_1} \right) \geq N_1 (1 + \rho_0/\rho_1) \tag{11.5}$$

and yields

$$\rho_0/\rho_1 \leq \frac{1 + \Delta\rho_1/\rho_1}{N_1 \dfrac{\Delta\rho_1}{\rho_1} - 1} \quad . \tag{11.6}$$

Condition (9.6) in Section 9 becomes

$$(N_1 - 1) \left(1 + \frac{\rho_0}{\rho_1 - \Delta\rho_2} \right) \geq N_1 (1 + \rho_0/\rho_1) \tag{11.7}$$

and yields

$$\rho_0/\rho_1 \geq \frac{1 - \Delta\rho_2/\rho_1}{N_1 \dfrac{\Delta\rho_2}{\rho_1} - 1} \tag{11.8}$$

Thus, any ρ_0 satisfying the condition

$$\frac{1 - \dfrac{\Delta\rho_2}{\rho_1}}{N_1 \dfrac{\Delta\rho_2}{\rho_1} - 1} \leq \rho_0/\rho_1 \leq \frac{1 + \Delta\rho_1/\rho_1}{N_1 \dfrac{\Delta\rho_1}{\rho_1} - 1} \quad , \tag{11.9}$$

will yield an optimal solution (N_1, ρ_1) for PROBLEM 1.

Let (N_1, ρ_1) now be an optimum solution to PROBLEM 2. Define $\Delta\rho_1 \geq 0$ and $\Delta\rho_2 \geq 0$ by

$$[N_B(N_1 + 1, \rho_1 + \Delta\rho_1)] = [N_B(N_1, \rho_1)] \quad , \tag{11.10}$$

$$[N_B(N_1 - 1, \rho_1 - \Delta\rho_2)] = N_B(N_1, \rho_1) \quad . \tag{11.11}$$

Equation (9.9) yields

$$\rho_1 + \Delta\rho_1 = \frac{(N_1 + 1)\,\rho_0'}{z_0 - (N_1 + 1)} \quad , \tag{11.12}$$

$$\frac{\rho_0'}{\rho_1} = (1 + \Delta\rho_1/\rho_1)\left(\frac{z_0}{N_1 + 1} - 1\right) \quad , \tag{11.13}$$

where ρ_0' is the upper limit for ρ_0 such that the optimal value for N remains N_1. Equation (9.9) also yields

$$\rho_1 - \Delta\rho_2 = \frac{(N_1 - 1)\,\rho_0''}{z - (N - 1)} \quad , \tag{11.14}$$

$$\frac{\rho_0''}{\rho_1} = \left(1 - \frac{\Delta\rho_2}{\rho_1}\right)\left(\frac{z_0}{N_1 - 1} - 1\right) \quad , \tag{11.15}$$

where ρ_0'' is the lower limit for ρ_0 such that the optimal value for N remains N_1. Thus, any ρ_0 satisfying the condition

$$\left(1 - \frac{\Delta\rho_2}{\rho_1}\right)\left(\frac{z_0}{N_1 - 1} - 1\right) \le \frac{\rho_0}{\rho_1} \le (1 + \Delta\rho_1/\rho_1)\left(\frac{z_0}{N_1 + 1} - 1\right) \tag{11.16}$$

will yield a value of N_1 in the optimal solution for PROBLEM 2. Note that the optimal value of ρ_1 varies according to

$$\rho = \frac{N\rho_0}{z_0 - N} \tag{11.17}$$

as the value of ρ_0 is varied.

Conditions (11.9) and (11.16) can be utilized to study the sensitivity of the optimal fractional investment level to the relative unit cost associated with the resupply system. Results of comparing the Poisson Model to the Parallel Server Model are given in Section 12.

SECTION 12

RESULTS AND CONCLUSIONS

Computer programs have been written and executed for

1) The Conservative Parallel Server Inventory Model described in Section 4,
2) The Poisson Model described in Section 5,
3) The Conservative Single Server Inventory Model described in Section 6.

Programs have also been written and executed for solving PROBLEM 1 and PROBLEM 2 described in Sections 8 and 9 utilizing the gradient technique described in Section 10 and procedures outlined in Section 9. Included in these programs are the sensitivity analyses given in Section 11. Program listings are found in Appendix A.

Reference material developed during the course of this project is given in the "Reference List and Bibliography" and in "Documents Internal to AFLC".

Various inputs were assumed to execute the computer programs to perform a detailed analysis and comparison of the behavior of the parallel server model (finite population) and the Poisson model (infinite population). Below are listed the major results.

- The Poisson demand process overestimates the backorder level for a given (N, ρ) specification.
- In general, the Poisson Model obtains a lower value for the optimum value of inventory investment than does the finite population model; as the system requirement m_0 increases, the difference between the optimum levels of inventory investment given by the two models becomes increasingly larger. Thus,

use of the Poisson Model underestimates optimum inventory investment level and results in a higher backorder level and a higher stockout probability than does the finite population level.

- If the relative unit cost (ρ_0) for resupply increases, then the difference between the optimum levels of inventory investment given by the two models becomes increasingly larger, approaching levels of 15-50%.
- In general, the finite population model given an optimum level for inventory investment which is less sensitive to a change in unit resupply cost than that suggested by the Poisson Model; i.e., the range for ρ_0 where the optimal solution remains invariant is larger for the finite population model than for the infinite population model.
- As the system requirement m_0 becomes larger, the optimum level for inventory investment becomes more sensitive to a change in unit resupply cost.

Some numerical outputs are tabulated in Appendix B.

The above results conclude that if the relative cost for resupply (manpower, equipment, administration, transportation, etc.) compared to inventory investment increases, then it becomes increasingly important that demand and resupply processes be accurately described to obtain cost effective allocations of expenditures between inventory investment levels (capital intensive) and resupply time levels (labor intensive). In addition, if the variance (uncertainty) of costs increases, then it is also important that correct descriptions of the demand and resupply processes be utilized in the decision making process. It is noted that inventory investment costs for recoverable items are well-known in comparison to projected costs for resupply during the life cycle of a typical weapons system. Thus, the fact that the finite population model tends to allocate a greater fraction of total expenditures to inventory investment than does the infinite population model makes the

finite population model even more attractive for adoption in the decision making process.

Non-optimal allocations that result when incorrect demand and resupply processes are assumed suggest the following areas for future investigation.

- Study resupply processes which allow for non-zero queue lengths (finite number of parallel servers).
- Generalize the trade-off analysis procedure developed in this report (resupply time level versus inventory investment level to minimize total system cost) to include interactions due to movement of many investment items in the transportation pipelines among user bases and supply depots.
- Further investigate demand and resupply modeling for recoverable items for purposes of determining transportation flows and optimal allocating of resources to inventory investment level and resupply time level.
- Investigate and compare the utility of the finite population model developed in this report with that of the Poisson model currently adopted by Air Force management [88, 89, 104] when incorporated into the METRIC and MOD-METRIC recoverable item control and allocation models.
- Investigate and compare the finite population model with the Poisson model in determining impacts on output of the Comprehensive Engine Management System (CEMS) being developed by the U.S. Air Force [D-11].

The above areas of investigation and those described in this report should contribute to a better understanding of certain problems of immediate concern to the U.S. Air Force. The questions dealt with here are very important for considerations that must be made when an integrated design effort is made to develop any proposed weapon system.

The work reported in this document was presented at two technical conferences which took place during the months of January and February of 1978. Particulars are given below.

1) "Analysis of Inventory/Resupply Models in Optimal Inventory Investment Allocation," ORSA/TIMS Special Interest Conference on Capital Budgeting," January 30, 31, and February 1, 1978, Phoenix, Arizona.
2) "Analysis of Inventory/Resupply Models in Optimal Materiel Distribution System Design," ORSA/TIMS/AIIE Special Interest Conference of Distribution Models, February 20-22, 1978. Hilton Head Island, South Carolina.

The above conferences were sponsored by the Operations Research Society of America (ORSA), The Institute of Management Sciences (TIMS), and the American Institute of Industrial Engineers (AIIE).

SECTION 13

REFERENCE LIST AND BIBLIOGRAPHY

1. Appelgran, Lief H., "A Column Generation Algorithm for a Ship Scheduling Problem," Transportation Science, III, No. 1, Feb. 1969, pp. 53-68.

2. Appelgren, Lief H., "Integer Programming Methods for a Vessel Scheduling Problem," Transportation Science, V. Feb. 1971, pp. 64-78.

3. Agin, Norman I. and Cullen, Daniel E., "An Algorithm for Transportation Routing and Vehicle Loading," Logistics, Vol. 1, Murray A. Geisler, Editor, North-Holland Publishing Co., 1975, pp. 1-20.

4. Balas, E., "An Additive Algorithm for Solving Linear Programs with Zero-One Variables," Ops. Res., Vol. 13, 1965, pp. 517-546.

5. Bellmore, M., Bennington, G., and Lubore, S., "A Multi-vehicle Tanker Scheduling Problem," Transportation Science, V, Feb. 1971, pp. 36-47.

6. Bellmore, M. and Malone, J. C., "Pathology of Traveling-Salesman Subtour-Elimination Algorithms," Ops. Res., Vol. 19, No. 2, Mar. - Apr., 1971, pp. 278-307.

7. Bellmore, M. and Nemhauser, G. L., "The Traveling Salesman Problem: A Survey," Ops. Res., Vol. 16, No. 3, 1968, pp. 538-558.

8. Bellmore, Mandell, Liebman, Jon, C., and Marks, David H., "An Extension of the (Swarc) Truck Assignment Problem," Naval Research Logistics Quarterly, Vol. 19, No. 1, Mar. 1972, pp. 91-99.

9. Benders, J. F., "Partitioning Procedures for Solving Mixed Variables Programming Problems," Numerische Mathematick, 4, 238-252 (1962).

10. Bennington, G. and Lubore, S., "Resource Allocation for Transportation," Naval Research Logistics Quarterly, Vol. 17, No. 4, Dec. 1970, pp. 471-484.

11. Blanning, Robert W., "Integrated Planning Systems," Office of Naval Research, Arlington, Virginia, Dec. 1971, 299 pages.

12. Burns, W. C., "Computation of Maximum Flows in Networks," Naval Post Graduate School, Monterey, California, June 1968, 60 pages.

13. Busacker, Robert G., Coffin, Stephen A., and Gowen, Paul J., "Three General Network Flow Problems and Their Solutions," Research Analysis Corp., Nov. 1962, McClean, Virginia 22101, 85 pages.

14. "Capability Design Specification, European Theater Network Analysis Model 1," McClean, Virginia: Research Analysis Corporation, April, 1969.

15. Cassidy, P. J. and Bennett, H. S., "TRAMP - A Multi-depot Vehicle Scheduling System," Operational Research Quarterly, Vol. 23, No. 2, pp. 151-163.

16. Christofides, Nicos and Eilon, Samuel, "An Algorithm for the Vehicle Dispatching Problem," Operational Research Quarterly, XX, Sept. 1969, pp. 309-310.

17. Clarke, G. and Wright, J. W., "Scheduling of Vehicles from a Central Depot to a Number of Delivery Points," Ops. Res., XII, July-Aug. 1964, pp. 568-581.

18. Conley, J. H., Farnsworth, R. S., Koenigsberg, E., Wiersema, J., "A Linear Programming Approach to the Total Movement of a Homogeneous Product," Transportation Science, II, Nov. 1968, pp. 289-309.

19. Cremeans, J. E., Smith, R. A., Tyndall, G. R., "Optimal Multicommodity Flows with Resource Allocation," Naval Research Logistics Quarterly, Vol. 17, No. 3, Sept. 1970, pp. 269-279.

20. Dantzig, George B. and Ramser, J. H., "The Truck Dispatching Problem," Management Science, VI, No. 1, Oct. 1959, pp. 80-91.

21. Dakin, R. J., "A Tree-Search Algorithm for Mixed Integer Programming Problems," Computer J., Vol. 8, 1965, pp. 250-255.

22. D'Espo, D. A., Dixon, H. L., Lefkowitz, B., "A Model for Simulating an Air Transportation System," Naval Research Logistics Quarterly, VII, Sept. 1960, pp. 213-220.

23. DeHayes, Daniel W., Jr., "Heuristics and Algorithms in the Solution of Physical Distribution Problems," Transportation J., IX, Summer 1970, pp. 41-47.

24. Elmaghraby, S. E. and Elshafei, A. N., "Branch and Bound Revisited: A Survey of Basic Concepts and Their Applications in Scheduling," Chapter 8, Modern Trends in Logistics Research, The MIT Press, Cambridge, Mass., 1976.

25. Elmaghraby, S. E., "Some Network Models in Management Science," Lecture Notes in Ops. Res. and Math. Systems, Springer-Verlag, New York, 1970.

26. Elmaghraby, S. E., "The Theory of Networks and Management Science: Part I," Management Science, XVII, Sept. 1970, pp. 25-32.

27. Erlander, S., "Transportation of Laundry - A Nonlinear Optimization Problem," Transpr. Res., Vol. 2, 1968, pp. 315-327.

28. Feeney, G. J. and Sherbrooke, C. C., "The (s-1, s) Inventory Policy Under Compound Poisson Demand," Management Science, Vol. 12, 1966, pp. 391-411; RM-4176-1-PR, The Rand Corp., Mar. 1966.

29. Ford, L. R. and Fulkerson, D. R., "A Suggested Computation for Maximal Multicommodity Network Flows," Mgmt. Sci., Vol. 5, 1958, pp. 97-101.

30. Garfinkel, R. A. and Nemhauser, G. L., Integer Programming, John Wiley, New York, 1972.

31. Garfinkel, R. S. and Nemhauser, G. L., "The Set Partitioning Problem: Set Covering with Equality Constraints," Ops. Res., Vol. 17, 1969, pp. 848-856.

32. Gaskell, T. J., "Bases for Vehicle Fleet Scheduling," Operational Research Quarterly, XVIII, Sept. 1967, pp. 281-295.

33. Gavish, Bezalel, "A Note on 'The Formulation of the M-Salesman Traveling Salesman Problem'," Management Science, Vol. 22, No. 6, Feb. 1976, pp. 704-706.

34. Gavish, B. and Schweitzer, P., "An Algorithm for Combining Truck Trips," Transportation Science, Vol. 8, No. 1, Feb. 1974, pp. 13-23.

35. Geoffrain, A. M., "A Guide to Computer Assisted Methods for Distribution Systems Planning," Working Paper No. 216, Western Management Science Institute, Univ. of California, Los Angeles, Sept. 1974, 42 pages.

36. Geoffrian, A. M., "Distribution Systems Configuration Planning: Case Study in the Application of a New Computer Based Method," Working Paper No. 219, Western Management Science Institute, Graduate School of Management, Univ. of California, Los Angeles, Calif., Sept. 1974, 40 pages.

37. Geoffrian, A. M., "Elements of Large Scale Math Programming," Management Science, Vol. 16, No. 11, 1970.

38. Geoffrian, A. M., "Generalized Benders Decomposition," J. of Opt. Theory and Appl., Vol. 10, No. 3, 1972, pp. 237-260.

39. Geoffrian, A. M., "Integer Programming by Implicit Enumeration and Balas' Method," The RAND Corp., RM-4783-PR, Feb. 1966.

40. Geoffrian, A. M. and Graves, G. W., "Multicommodity Distribution System Design by Benders Decomposition," Management Science, Vol. 20, No. 5, Jan. 1974, pp. 822-844.

41. Gillett, Billy E. and Miller, Leland T., "A Heuristic Algorithm for the Vehicle Dispatch Problem," Ops. Res., Vol. 22, No. 2, Mar. - Apr. 1974, pp. 340-349.

42. Glover, F., "A Multiphase-Dual Algorithm for the Zero-One Integer Programming Problems," Ops. Res., Vol. 13, 1965, pp. 497-520.

43. Glover, F., "Surrogate Constraints," Ops. Res., Vol. 16, 1968, pp. 741-749.

44. Gomory, R. E., "An Algorithm for the Mixed Integer Problem," RAND Report P1885, Feb. 1960.

45. Gomory, R. E. and Hu, T. C., "Synthesis of a Communication Network," J. SIAM, Vol. 12, #2, 1964, pp. 348-369.

46. Gould, F. J., "A Linear Programming Model for Cargo Movement Evaluation," Transportation Science, Vol. 5, No. 4, Nov., 1971, pp. 344-365.

47. Gould, J., "The Size and Composition of a Road Transport Fleet," Operational Research Quarterly, Vol. 20, No. 1, pp. 81-92.

48. Gross, Donald and Soriano, A., "On the Economic Application of Airlift to Product Distribution and Its Impact on Inventory Levels," Volume 19, No. 3, Sept. 1972, pp. 501-507.

49. Gunn, William A., "Airline System Simulation," Ops. Res., XII, Mar. - Apr. 1964, pp. 206-229.

50. Hausman, Warren H. and Gilmour, Peter, "A Multi-Period Truck Delivery Problem," Transportation Research, Vol. 1, pp. 349-357.

51. Hipp, William S., "An Algorithm for Rapid Reassignment of Motor Carriers," Software Age, II, Nov. 1968, pp. 8, 9, 12 and 14.

52. Holmes, R. A. and Parker, R. G., "A Vehicle Scheduling Procedure Based Upon Savings and a Solution Pertubation Scheme," Operational Research Quarterly, Vol. 27, 1, i., pp. 83-92.

53. Horowitz, Harold M., "Mathematical Modeling for Transportation Planning," The Logistics and Transportation Review, Univ. of British Columbia, Vol. 7, No. 31, Mar. 1971, 7 pages.

54. Hu, T. C., Integer Programming and Network Flows, Addison-Wesley, 1969, pp. 255-271.

55. Hunt, Donald L. and Wilkerson, W. L., "Bounds for Maximal Temporally Repeated Flows in a Network," T-204, The George Washington University, Washington, D. C., Mar. 1968, 94 pages.

56. Hyman, Warren and Gordon, Larry, "Commercial Airline Scheduling Techniques," Transportation Research, II, March 1968, pp. 23-29.

57. Kleinrock, Leonard, Queueing Systems, Volume I: Theory, John Wiley, New York, 1975.

58. Knight, K. W. and Hofer, J. P., "Vehicle Scheduling with Timed and Connected Calls: A Case Study," Operational Research Quarterly, XIX, No. 3, Sept. 1968, pp. 299-310.

59. Koch, Barry S., "Simulator for Transportation Analysis and Planning (SITAP), System Description," National Military Command System Support Center, Washington, D. C. 20301, Nov. 68, 224 pages.

60. Kovacs, A. Z., "Visit Allocation Model, Vol. I, Algorithm Description," Technical Report UDRI-TR-70-32, Univ. of Dayton Research Institute, ASD(ASFR), U. S. Air Force, WPAFB, Ohio, Report No. ASB-70-8, July 1970, 23 pages.

61. Kovacs, A. Z., "Visit Allocation Model, Vol. II, Program Description," Technical Report UDRI-TR-70-73, Univ. of Dayton Research Institute, ASD(ASFR), U. S. Air Force, WPAFB, Ohio, Report No. ASB-70-9, July 1970, 130 pages.

62. Kovacs, A. Z., "Strategic Penetration Path Algorithm," Proceedings of the 26th Military Operations Research Symposium at the Naval Post Graduate School, Monterey, California, Nov. 1970, pp. 175-182.

63. Kovacs, A. Z., "Visit Allocation Model, Vol. III, Algorithm Modifications," Technical Report UDRI-TR-72-36, Univ. of Dayton Research Institute, ASD(XRO), U. S. Air Force, WPAFB, Ohio, Report No. R-72-6, June 1972, 93 pages.

64. Kovacs, A. Z., "Visit Allocation Model, Vol. IV, Program Instruction Manual," Technical Report UDRI-TR-72-49, Univ. of Dayton Research Institute, ASD(XRO), U. S. Air Force, WPAFB, Ohio, Report No. R-73-3, Nov. 1972, 101 pages.

65. Kovacs, A. Z., "Spherical Path Algorithms," to be published as a Technical Report, Univ. of Dayton Research Institute, 1974, Submitted to Directorate of Advanced Systems Analysis, Aeronautical Systems Division (XRO), WPAFB, Ohio, 139 pages.

66. Krolak, Patrick, Felts, Wayne, and Nelson, James, "A Man-Machine Approach Toward Solving the Generalized Truck Dispatching Problem," Transportation Science, VI, May 1972, pp. 149-170.

67. Laderman, J., Gleiberman, L., Egan, J. F., "Vessel Allocation by Linear Programming," Naval Research Logistics Quarterly, XII, Sept. 1966, pp. 315-320.

68. Lam, Tenny N., "Comments on a Heuristic Algorithm for the Multiple Terminal Problem," Transportation Science, IV, Nov., 1970, pp. 403-405.

69. Lampkin, W. and Saalmans, P. D., "The Design of Routes, Service Frequencies, and Schedules for a Muncipal Bus Undertaking: A Case Study," Operational Research Quarterly, Vol. 18, No. 4, pp. 375-397.

70. Land, A. H. and Doig, A. G., "An Automatic Method of Solving Discrete Programming Problems," Econometrica, Vol. 28, 1960, pp. 497-520.

71. Langley, Robert W., Kennington, Jeff, Shetty, C. M., "Efficient Computational Devices for the Capacitated Transportation Problem," Naval Research Logistics Quarterly, Vol. 21, No. 4, Dec. 1974, pp. 637-647.

72. Lasdon, L. S., Optimization Theory for Large Systems, The Macmillan Co., New York, 1970.

73. Lenstra, J. K., "Clustering Data Array and The Traveling Salesman Problem," Ops. Res., Vol. 22, No. 2, Mar. - Apr. 1974, pp. 413-414.

74. Levin, Amos, "Scheduling and Fleet Routing Models for Transportation Systems," V, Aug. 1971, pp. 232-255.

75. Lin, S. and Kernighan, B. W., "An Effective Heuristic Algorithm for the Traveling Salesman Problem," Ops. Res. Vol. 21, No. 2, Mar. - Apr. 1973, pp. 498-516.

76. Little, J. D. C., Murty, K, G., Sweeney, D. W., and Karel, C., "An Algorithm for the Traveling Salesman Problem," Ops. Res., Vol. 11, 1963, pp. 972-989.

77. Manheim, Marvin L., "Principles of Transport Systems Analysis," P67-1, Massachusetts Institute of Technology, Dept. of Civil Engineering, Jan. 1967, 14 pages.

78. Marlow, W. H. (ed.), Modern Trends in Logistics Research, The MIT Press, Cambridge, Mass., 1976.

79. McKay, M. D. and Hartley, H. O., "Computerized Scheduling of Seagoing Tankers," Vol. 21, No. 2, Naval Research Logistics Quarterly, June 1974, pp. 255-264.

80. McPherson, Michael F. and O'Hara, Brian, "A Computer Assisted Method for Determining LOGAIR Route Structures," Master's Thesis, Graduate Education Division, School of Systems and Logistics, Air Force Institute of Technology, WPAFB, Ohio, SLSR 17-76A, June 1976, 70 pages.

81. Midler, J. L. and Wollmer, R. D., "A Flight Planning Model for the Military Airlift Command," The Rand Corp., RM-5722-PR, Santa Monica, California, Oct. 1968.

82. Midler, J. L., "A Stochastic Multiperiod Multimode Transportation Model," Transportation Science, Vol. 3, No. 1, 1969, pp. 8-29.

83. Miller, C. E., Tucker, A. W., and Zemlin, "Integer Programming Formulation of Traveling Salesman Problems," J. Ass. Comput. Mach., Vol. 7, 1960, pp. 326-329.

84. Mitten, L. G., "Branch-and-Bound Methods: General Formulation and Properties," Ops. Res., Vol. 18, 1970, pp. 24-34.

85. Moberly, Kenton D. and Gorychka, Theordore C., "A Mathematical Model for Determining LOGAIR Flight Schedules," Master's Thesis, Graduate Education Division, School of Systems and Logistics, Air Force Institute of Technology, QPAFB, Ohio, SLSR 27-76A, June 1976, 90 pages.

86. Morse, Phillip M., Queues, Inventories and Maintenance, The Analysis of Operational Systems with Variable Demands and Supply, John Wiley, 1963.

87. Mossman, Frank H. and Hynes, James P., Systems Network Theory: Applications to Distribution Problems, D. H. Mark Publishing Co., Braintree, Massachusetts, 1968.

88. Muckstadt, John A. and Pearson, John M., "Mod-Metric: A Multi-Item, Multi-Echelon, Multi-Indenture Inventory Model," School of Systems and Logistics Technical Report, Air University, Air Force Institute of Technology, WPAFB, Ohio, Internal SL Technical Report, Feb. 1976, 36 pages.

89. Muckstadt, John A., "A Model for a Multi-Item, Multi-Echelon, Multi-Indenture Inventory System," Management Science, Vol. 20, No. 4, Part I, Dec. 1973, pp. 472-481.

90. Nace, G. E., "Computerized Truck Scheduling," Distribution Manager, LXVIII, Mar. 1969, pp. 37-42.

91. Nace, G. E., "Distributing Goods by VSP/360," Software Age, III, Mar. 1969, pp. 8, 9, 11-16, 20, 21, 26, and 29.

92. Nemhauser, G. L. and Yu, Po-Lung, "Bulk Service Scheduling," Core Discussion Papers, No. 7017, Center for Operations Research and Econometrics, International Center for Management Sciences, CORE, Belgium.

93. Noonan, R. and Whinston, A., "An Information System for Vehicle Scheduling," Software Age, III, Dec. 1969, pp. 8-13.

94. Palmatier, Philip E. and Prescott, Gary T., "The Feasibility of Developing and Solving a Model to Determine LOGAIR Routing Structures," Report No. SLSR330-75B, Master's Thesis, Graduate Education Division, School of Systems and Logistics, Air Force Institute of Technology, WPAFB, Ohio, Aug. 1975, 90 pages.

95. Panagiotakopoulos, D., "Multi-Commodity Multi-Transformed Network Flows with an Application to Residuals Management," Management Science, Vol. 22, No. 8, Apr. 1976, pp. 874-882.

96. Pierce, J. F., "Direct Search Algorithms for Truck-Dispatching Problems, Part I," Transportation Research, Vol. 3, 1969, pp. 1-42.

97. Rao, M. R. and Zionts, S., "Allocation of Transportation Units to Alternate Trips-A Column Generation Scheme with Out-of-Kilter Subproblems," Ops. Res., XVI, Jan. - Feb. 1968, pp. 52-53.

98. Richardson, R., "An Optimization Approach to Routing Aircraft," Trans. Sci., Vol. 10, No. 1, Feb. 1976, pp. 52-71.

99. Roelofs, Thomas B., Throop, Thomas A., Tyndall, Gene R., "Simulation and Gaming Methods for Analysis of Logistics (SIGMALOG): Network Analysis Model," Research Analysis Corp., RAC-TP-353, May 1969, 117 pages.

100. Ruefli, Timothy W., "Decentralized Transshipment Networks," Ops. Res., XIX, Nov. - Dec. 1971, pp. 1619-1631.

101. Schaefer, Margaret K. and Hurter, Arthur P., "An Algorithm for the Solution of a Location Problem with Metric Constraints," Naval Research Logistics Quarterly, Vol. 21, No. 4, Dec. 1974, pp. 625-636.

102. Schuster, A. D., and LaLonde, B. J., "Twenty Years of Distribution Modeling," Unpublished Report, The Ohio State University.

103. Seshagiri, N., Narasimhan, R., Mehndiratta, L., and Chanda, B. K., "Computer Generated Time-Tables and Bus Schedules for a Large Bus Transport Network," Transportation Science, Vol. 3, No. 1, Feb. 1969, pp. 69-85.

104. Sherbrooke, Craig C., "Metric: A Multi-Echelon Technique for Recoverable Item Control," Ops. Res., Vol. 16, 1968, pp. 122-141; Memorandum RM-5078-PR, The Rand Corp., Nov. 1966, 55 pages.

105. "The Structure of Transportation Networks," U.S. Army Transportation Research CMD, Fort Eustis, Virginia, TR-62-11, May 1962, 115 pages.

106. Svestka, Joseph A. and Huckfeldt, Vaughn E., "Computational Experience with an M-Salesman Traveling Salesman Algorithm," Management Science, Vol. 19, No. 7, Mar. 1973, pp. 79-799.

107. Szwarc, Wlodzimierz, "The Truck Assignment Problem," Naval Research Logistics Quarterly, XIV, Dec. 1967, pp. 529-557.

108. Taha, H. A., Operations Research, MacMillan Publishing Co., New York, 1971.

109. Tillman, Frank A., "The Multiple Terminal Delivery Problem with Probabilistic Demands," Transportation Science, Vol. 3, No. 3, Aug. 1969, pp. 192-204.

110. Tillman, Frank A., "The Author's Reply-A Note on the Multiple Terminal Problem with Probabilistic Demands," Transportation Science, IV, May 1970, pp. 232-237.

111. Tillman, Frank A. and Cochran, Harold, "A Heuristic Approach for Solving the Delivery Problem," The J. of Industrial Engineering, Vol. XIX, No. 7, July 1968, pp. 354-358.

112. Tillman, Frank A. and Hering, Robert W., "A Study of a Look-Ahead Procedure for Solving the Multiterminal Delivery Problem," Transportation Research, Vol. 5, 1971, pp. 225-229.

113. Tomlin, J. A., "Minimum Cost Multicommodity Network Flows," Ops. Res. XIV, Jan. - Feb. 1966, pp. 45-51.

114. Turner, W. C. and Houghland, E. S., "The Optimal Routing of Solid Waste Collection Vehicles," AIIE Transactions, Vol. 7, Nov. 4, Dec. 1975, pp. 427-431.

115. Turner, Wayne C., Ghare, Prabhakar, M., Fourds, Leslie R., "Transportation Routing Problem--A Survey," AIIE Transactions, Vol. 6, No. 4, Dec. 1974, pp. 288-301.

116. Uebe, Gotz, "Comments on 'The Multiple Terminal Problem with Probabilistic Demands'," Transportation Science, IV, May 1970, pp. 226-231.

117. United States Department of Defense, Transportation Movement Planning System (TRAMPS), Analyst's Guide, Washington, D. C.: Defense Communications Agency, June 30, 1967.

118. United States Department of Defense, Transportation Movement Planning System (TRAMPS), System Description, Washington, D. C.: Defense Communications Agency, June 30, 1967.

119. Walker, Warren, "Using the Set-Covering Problem to Assign Fire Companies to Fire Houses," Ops. Res., Vol. 22, No. 2, Mar. - Apr. 1974, pp. 275-277.

120. Webb, M. H. J., "Relative Performance of Some Sequential Methods of Planning Multiple Delivery Journeys," Operational Research Quarterly, Vol. 23, No. 3, Sept. 1972, pp. 361-372.

121. Weigel, Henry S., Cremeans, John E., "The Multicommodity Network Flow Model Revised to Include Vehicle Per Time Period and Node Constraints," Naval Research Logistics Quarterly, Vol. 19, No. 1, Mar. 1972, pp. 77-89.

122. Wilkinson, "An Algorithm for Universal Maximal Dynamic Flows in a Network," Ops. Res., XIX, Nov. - Dec. 1971, pp. 1602-1612.

123. Williams, A. C., "A Stochastic Transportation Problem," Ops. Res., XI, Sept. - Oct. 1963, pp. 759-770.

124. Wren, Anthony and Holiday, Alan, "Computer Scheduling of Vehicles from One or More Depots to a Number of Delivery Points," Operational Research Quarterly, XXIII, No. 3, Sept. 1972, pp. 333-344.

125. Zacks, S., "A Two Echelon Multi-Station Inventory Model for Navy Applications," Vol. 17, No. 1, Naval Research Logistics Quarterly, Mar. 1970, pp. 79-85.

SECTION 14

DOCUMENTS INTERNAL TO AFLC

D-1. "Air Logistics Center Shipment Planning--A New Look at Transportation,"--Air Force briefing.

D-2. "Air Logistics Pipeline Study (ALPS)," Office of the Assistant Secretary of Defense, Installations and Logistics, Jan. 1976.

D-3. "Cargo Shipments Table," origin to destination matrices, 1976.

D-4. "Civil Aeronautics Board Coordinates Sequenced Ascending by City Name,"--longitude, latitude coordinates for cities and distances between cities.

D-5. Darter, William S., "Fill Rate Study," AFLC/DSTMO, 1973, 3 pages.

D-6. Darter, William S., "In Consideration of Logex," AFLC/DSTMO.

D-7. Darter, William S., "Proposal for Investment Optimal Mode Selection," AFLC/DSTMO.

D-8. Fitzgerald, John, "Logistics Analysis Draft, Gamma Forecast Error Model," AFLC/AQMLE, 21 November 1975, 8 pages.

D-9. Fitzgerald, John, "Logistics Analysis Draft, Three Echelon LRU Search Algorithm," AFLC/AQMLE, 1 October 1975, 12 pages.

D-10. Fitzgerald, John, "Resupply Time Convexity, Three Echelon Draft Addendum," AFLC/AQMLE, 8 January 1976, 6 pages.

D-11. "Functional Description for Comprehensive Engine Management System (CEMS)," FD-D-10037-A, AFLC/ADQ(2), 30 April 1976, 159 pages.

D-12. "FY 77 Funding of LOGAIR Model," Memorandum to AFLC/XRS from Colonel Bob M. Pennington, DCS/Distribution, 27 Jan. 1976.

D-13. "Goals/Purposes of LOGAIR System,"--proposed LOGAIR route structure, FY 77.

D-14. "Impact of Transportation Times on Inventory Requirements,"--Air Force briefing.

D-15. Lt. Karenbauer and Mary Hillis, "Description of the LOGAIR Problem," ASD/AFOSR, 14 July 1975.

D-16. "LOGAIR Route Structure," current, Jund 1976.

D-17. "LOGAIR Schedule FY 76, Revision Seven," 25 April 1976.

D-18. "Mode of Shipment Codes," Chapter 1, DOD 4500.32-R, Vol. I, page B-94.

D-19. Tars Guide.

D-20. "Time Standards," from Uniform Materiel Movement and Issue Priority System, DOD DIR 4410.6, Sept. 1975.

D-21. "United States Air Force Logistics Airlift, FY 76, Flight Schedules and Routing Guide," 1 July 1975.

APPENDIX A

PROGRAM LISTINGS

CONSERVATIVE PARALLEL SERVER
INVENTORY MODEL
(FINITE POPULATION)
PROGRAM LISTING

```
/LOGON  "NC"
/DO RUN.WATFIV
$JOB (L8NTJQQ)        KOVACS
      DOUBLE PRECISION P(101),PA(101),DER2(101),NB,NB1,PO,PO1,Z,NTOT,RO
     1M1,MO,ANS19,DER1,DER3,VAR,VARA,NBSQ,VARN,RESPSQ,VARS,SN,MN,VARM,U
     2SSQ,MNSQ,T
      Z=0.
      PRINT5
5     FORMAT('1',5X,'***KOVACS FININITE SERVER INVENTORY MODEL***')
      READ(5,*) NTOT,RO,MO,M1
      PRINT10,NTOT,RO,MO,M1
10    FORMAT(' NTOT=',D20.14,' RO=',D20.14,' MO=',D20.14,' M1=',D20.14)
      ITOT=NTOT
      PRINT150
      IX=IY=0
      CALL NINE(NTOT,MO,P,PO,RO,IX,IY)
      CALL TEN(NTOT,MO,P,PO,RO,IX,IY)
C IX AND IY ARE TOTALS FOR OVERFLOWS AND UNDERFLOWS IN THE PROGRAM.
      PRINT220,IX,IY
150   FORMAT('-P(N) IS THE PROBABILITY THAT N ITEMS ARE IN RESUPPLY AST
     1IME NEARS INFINITY')
C T IS THE TOTAL OF THE PROBABILITIES -- SHOULD EQUAL 1.00.
      T=PO
      DO 100 I=1,ITOT
100   T=T+P(I)
140   FORMAT('OTHE EXPECTED NUMBER OF BACKORDERS IS ',D20.14)
```

```
160   FORMAT('0THE TOTAL OF ALL THE P VALUES IS ',D20.14)
      PRINT160,T
      CALL SIXTEN(NTOT,M1,P,NB)
      PRINT140,NB
      I6=NTOT-M1+1
      IF(I6.LE.0)Z=P0
      IF(I6.LE.0)I6=1
      DO 300 I7=I6,ITOT
300   Z=Z+P(I7)
C Z IS THE TOTAL OF ALL THE PROBABILITIES FROM I=NTOT-M1+1 TO I=NTOT.
      PRINT301,Z
301   FORMAT('0THE PROBABILITY FOR STOCKOUT IS ',D20.14)
      IX=IY=0
      CALL NINE(NTOT+1,M0,PA,P01,R0,IX,IY)
      CALL TEN(NTOT+1,M0,PA,P01,R0,IX,IY)
      I79=ITOT+1
      PRINT220,IX,IY
220   FORMAT('0THERE WERE ',I3,' UNDERFLOWS AND ',I3,' OVERFLOWS')
      T=P01
      DO 102 I=1,I79
102   T=T+PA(I)
      PRINT160,T
      CALL SIXTEN(NTOT+1,M1,PA,NB1)
      PRINT140,NB1
      PRINT142,NB-NB1
142   FORMAT(' DELTA NB=',D20.14)
      PRINT400,ITOT,ITOT+1,ITOT,+(NB-NB1)/NB
```

```
400   FORMAT('0NB(',I3,') - NB(',I3,') / NB(',I3,') =',D20.14)
      CALL SUB21(NTOT,P,DER1)
      PRINT310,DER1
310   FORMAT('0THE EXPECTED NUMBER OF UNITS IN RESUPPLY IS ',D20.14)
      PRINT320,DER1/(0.0-RO)
320   FORMAT('0THE RESULT OF FORMULA 21 IS ',D24.14)
      CALL SUB20(NTOT,RO,DER1,DER2)
      PRINT333
      PRINT330,0,DER1/(0.-RO)
      PRINT330,(I,DER2(I),I=1,ITOT)
333   FORMAT('0ANSWERS FOR FORMULA 20:')
330   FORMAT(' P(',I3,') GIVES ',D24.14)
      CALL VAR1(PO,M1,NTOT,VAR,P,NB)
      CALL VAR1(PO1,M1,NTOT+1,VARA,PA,NB1)
      PRINT350,ITOT,VAR
350   FORMAT(' THE VARIANCE OF NB FOR NTOT=',I3,' IS ',D20.14)
      PRINT350,ITOT+1,VARA
      DER3=0.0
      CALL VAR1(PO,M1,NTOT,NBSQ,P,DER3)
      CALL SUB22(NTOT,M1,NB,DER1,DER3,RO,NBSQ)
      PRINT360,DER3
360   FORMAT(' THE ANSWER TO FORMULA 22 IS ',D20.14)
      PRINT352,NBSQ
      CALL VAR2(DER1,P,VARN,PO,NTOT)
352   FORMAT('0THE EXPECTED VALUE OF THE SQUARE OF THE BACKORDER LEVEL I
     1S:',D20.14)
      PRINT410,VARN
```

```
410   FORMAT('OTHE VARIANCE OF THE EXPECTED NUMBER OF UNITS IN RESUPPLY
     1IS:',D20.14)
      DER3=0.0
      CALL VAR2(DER3,P,RESPSQ,PO,NTOT)
      PRINT420,RESPSQ
420   FORMAT('OTHE EXPECTED VALUE OF THE SQUARE OF THE NUMBER OF UNITSI
     IN RESUPPLY IS:',D20.14)
      CALL SNAVE(NTOT,M1,P,PO,SN)
      PRINT430,SN
430   FORMAT('OTHE EXPECTED STOCK LEVEL IS ',D20.14)
      CALL VAR3(NTOT,M1,P,PO,VARS,SN)
      PRINT440,VARS
440   FORMAT('OTHE VARIANCE OF THE STOCK LEVEL IS ',D20.14)
      U=0.0
      CALL VAR3(NTOT,M1,P,PO,SSQ,U)
425   FORMAT('OTHE EXPECTED VALUE OF THE SQUARE OF THE STOCK LEVEL IS:'
     1D20.14)
      PRINT425,SSQ
      CALL MNAVE(NTOT,M1,P,PO,MN)
      PRINT435,MN
435   FORMAT('OTHE EXPECTED NUMBER IN SERVICE IS ',D20.14)
      CALL VAR4(NTOT,M1,MN,P,VARM,PO)
      PRINT444,VARM
444   FORMAT('OTHE VARIANCE OF THE EXPECTED NUMBER IN SERVICE IS ',D20.
     14)
      CALL VAR4(NTOT,M1,U,P,MNSQ,PO)
      PRINT446,MNSQ
```

```
446   FORMAT('0THE EXPECTED VALUE OF THE SQUARE OF THE NUMBER IN SERVIC
     1 IS:',D20.14)
      STOP
      END
      SUBROUTINE NINE(NTOT,MO,P,PO,RO,IX,IY)
C FINDS THE PROBABILITIES FOR J=0 TO J=NTOT-MO (FINDS P(J)).
C USES NATURAL LOGS AND EXPONENTS TO ALLOW FOR LARGER SYSTEMS
      DOUBLE PRECISION P(101),NTOT,PO,MO,RO,A,B,C,D,E,F,G,H,Z,Y
      DOUBLE PRECISION DLOG,DEXP
      ITEST=NTOT-MO+1
      DO 10 J1=1,ITEST
      Y=Z=0.0
      J=J1-1
      DO 20 N1=1,ITEST
      NRESUP=N1-1
      A=(NRESUP-J)*DLOG(MO*RO)
      C=D=0.0
      IF(J.LE.1)GOTO 35
      DO 30 K=2,J
      B=K
30    C=C+DLOG(B)
35    IF(NRESUP.LE.1)GOTO 45
      DO 40 K=2,NRESUP
      B=K
40    D=D+DLOG(B)
45    E=A+C-D
      CALL FLOW(E,J,IX,IY)
      Z=Z+DEXP(E)
```

```
20    CONTINUE
      ITOT=NTOT
      DO 50 NRESUP=ITEST,ITOT
      C=D=0.0
      M=MO
      IF(M.LE.1)GOTO 65
      DO 60 K=2,M
      B=K
60    C=C+DLOG(B)
65    IF(J.LE.1)GOTO 75
      DO 70 K=2,J
      B=K
70    D=D+DLOG(B)
75    A=F=0.0
      IF(NRESUP.LE.1)GOTO 85
      DO 80 K=2,NRESUP
      B=K
80    A=A+DLOG(B)
85    N=NTOT-NRESUP
      IF(N.LE.1)GOTO 95
      DO 90 K=2,N
      B=K
90    F=F+DLOG(B)
95    G=(NTOT-MO-J)*DLOG(MO)
      H=(NRESUP-J)*DLOG(RO)
      E=C+D+G+H-A-F
      CALL FLOW(E,J,IX,IY)
      Y=Y+DEXP(E)
```

```
50      CONTINUE
        IF(J.EQ.0)P0=1.0/(Y+Z)
        IF(J.EQ.0)PRINT503,J,P0
        IF(J.EQ.0)GOTO 10
        P(J)=1./(Y+Z)
        PRINT503,J,P(J)
503     FORMAT(' P(',I3,') = ',D20.14)
10      CONTINUE
        RETURN
        END
        SUBROUTINE TEN(NTOT,MO,P,PO,RO,IX,IY)
C FINDS THE PROBABILITIES FOR J=NTOT-MO+1 TO J=NTOT (USES NATURALDLOGST
O ALLOW FOR LARGER NTOT).
        DOUBLE PRECISION P(101),NTOT,PO,MO,RO,A,B,C,D,E,F,G,H,Z,Y
        DOUBLE PRECISION DLOG,DEXP
        INITAL=NTOT-MO+1
        ITOT=NTOT
        DO 10 J=INITAL,ITOT
        Y=Z=0.0
        DO 20 N1=1,INITAL
        NRESUP=N1-1
        A=C=H=G=0.0
        IF(J.LE.1)GOTO 105
        DO 100 K=2,J
        B=K
100     G=G+DLOG(B)
```

AD-A055 120 DAYTON UNIV OHIO SCHOOL OF ENGINEERING F/G 15/5
ANALYSIS OF INVENTORY/RESUPPLY SUB-SYSTEM MODELS IN OPTIMAL MAT--ETC(U)
APR 78 A Z KOVACS AFOSR-77-3266
UNCLASSIFIED UDR-TR-78-40 NL

2 OF 2

AD A055120

END
DATE
FILMED
7-78
DDC

```
105    N=NTOT-J
       IF(N.LE.1)GOTO 115
       DO 110 K=2,N
       B=K
110    H=H+DLOG(B)
115    IF(NRESUP.LE.1)GOTO 35
       DO 30 K=2,NRESUP
       B=K
30     A=A+DLOG(B)
35     N=MO
       IF(N.LE.1)GOTO 45
       DO 40 K=2,N
       B=K
40     C=C+DLOG(B)
45     D=(NRESUP-J)*DLOG(RO)
       E=(NRESUP-(NTOT-MO))*DLOG(MO)
       F=D+E+G+H-A-C
       CALL FLOW(F,J,IX,IY)
       Y=Y+DEXP(F)
20     CONTINUE
       DO 50 NRESUP=INITAL,ITOT
       A=C=D=E=F=0.0
       IF(J.LE.1)GOTO65
       DO 60 K=2,J
       B=K
60     A=A+DLOG(B)
```

```
65      I=NTOT-J
        IF(I.LE.1)GOTO 75
        DO 70 K=2,I
        B=K
70      C=C+DLOG(B)
75      I=NTOT-NRESUP
        IF(I.LE.1)GOTO 85
        DO 80 K=2,I
        B=K
80      E=E+DLOG(B)
85      IF(NRESUP.LE.1)GOTO 95
        DO 90 K=2,NRESUP
        B=K
90      F=F+DLOG(B)
95      D=(NRESUP-J)*DLOG(RO)
        B=D+A+C-E-F
        CALL FLOW(B,J,IX,IY)
        Z=Z+DEXP(B)
50      CONTINUE
        P(J)=1.0/(Y+Z)
        PRINT506,J,P(J)
506     FORMAT(' P(',I3,') = ',D20.14)
10      CONTINUE
        RETURN
        END
```

```
      SUBROUTINE SIXTEN(NTOT,M1,P,NB)
C FINDS THE DEXPECTED NUMBER OF BACKORDERS.
      DOUBLE PRECISION P(101),NB,PO,NTOT,M1,B
      ITOT=NTOT
      NB=0.
      INITAL=NTOT-M1+1
      DO 10 NRESUP=INITAL,ITOT
      B=(M1-(NTOT-NRESUP))*P(NRESUP)
      PRINT20,NRESUP,B
20    FORMAT(' NB(',I3,') = ',D20.14)
10    NB=NB+B
      RETURN
      END
      SUBROUTINE SUB20(NTOT,RO,DER1,DER2)
C FINDS THE RESULTS FOR FORMULA 20.
      DOUBLE PRECISION NTOT,RO,DER1,DER2(101)
      ITOT=NTOT                                                 7  2351
      DO 10 NRESUP=1,ITOT
10    DER2(NRESUP)=NRESUP/RO-DER1/RO
      PODER=DER1/(0.0-RO)
      RETURN
      END
      SUBROUTINE SUB21(NTOT,P,DER1)
C FINDS THE DEXPECTED NUMBER OF UNITS IN RESUPPLY.
      DOUBLE PRECISION NTOT,P(101),DER1
      DER1=0.0
      ITOT=NTOT
      DO 10 NRESUP=1,ITOT
```

```
10    DER1=NRESUP*P(NRESUP)+DER1
      RETURN
      END
      SUBROUTINE SUB22(NTOT,M1,NB,DER1,DER2,RO,NBSQ)
C FINDS THE RESULTS FOR FORMULA 22.
      DOUBLE PRECISION NTOT,M1,NB,DER1,DER2,RO,NBSQ
      DER2=NB/RO*((NTOT-M1)-DER1+NBSQ/NB)
      RETURN
      END
      SUBROUTINE VAR3(NTOT,M1,P,PO,VAR,SN)
      DOUBLE PRECISION NTOT,M1,P(101),PO,VAR,SN
      ITEST=NTOT-M1
      VAR=PO*(NTOT-M1-SN)**2
      DO 10 NRESUP=1,ITEST
10    VAR=VAR+P(NRESUP)*(NTOT-M1-NRESUP-SN)**2
      RETURN
      END
      SUBROUTINE SNAVE(NTOT,M1,P,PO,SN)
      DOUBLE PRECISION NTOT,M1,P(101),PO,SN
      ITEST=NTOT-M1
      SN=PO*(NTOT-M1)
      DO 10 NRESUP=1,ITEST
      SN=SN+P(NRESUP)*(NTOT-M1-NRESUP)
10    CONTINUE
      RETURN
      END
```

```
      SUBROUTINE MNAVE(NTOT,M1,P,PO,MN)
      DOUBLE PRECISION NTOT,M1,P(101),PO,MN
      INITAL=NTOT-M1
      MN=0
      IF(INITAL.EQ.0)MN=PO*(NTOT)
      ITOT=NTOT
      IF(INITAL.EQ.0)INITAL=1
      DO 10 NRESUP=INITAL,ITOT
10    MN=MN+(NTOT-NRESUP)*P(NRESUP)
      MN=MN+M1*PO
      ITEST=INITAL-1
      DO 20 NRESUP=1,ITEST
20    MN=MN+M1*P(NRESUP)
      RETURN
      END
      SUBROUTINE VAR4(NTOT,M1,MN,P,VAR,PO)
      DOUBLE PRECISION NTOT,M1,MN,P(101),VAR,PO
      INITAL=NTOT-M1
      ITEST=INITAL+1
      VAR=PO
      DO 20 NRESUP=1,ITEST
20    VAR=VAR+P(NRESUP)
      VAR=VAR*(M1-MN)**2
      IF(INITAL.EQ.0)VAR=VAR+PO*(NTOT-MN)**2
      IF(INITAL.EQ.0)INITAL=1
      ITOT=NTOT
      DO 10 NRESUP=INITAL,ITOT
```

```
10      VAR=VAR+P(NRESUP)*(NTOT-NRESUP-MN)**2
        RETURN
        END
        SUBROUTINE FLOW(X,J,IX,IY)
C CHECKS FOR OVERFLOW AND UNDERFLOW --PRINTS WARNING MESSAGE IF ONE ISF
OUND.
        DOUBLE PRECISION X
        IF(X.LT.-174.)IX=IX+1
        IF(X.GT.174.)IY=IY+1
        IF(X.LT.-174.)PRINT5,J
5       FORMAT(' *****UNDERFLOW*****  J=',I3)
        IF(X.LT.-174.)X=-174.
        IF(X.GT.174.)PRINT10,J
10      FORMAT(' *****OVERFLOW*****  J=',I3)
        IF(X.GT.174.)X=174.
C SETS X EQUAL TO MAX OR MIN VALUE SO OVERFLOW OR UNDERFLOW DOESN'T OCC
R IN DEXPONENT.
        RETURN
        END
        SUBROUTINE VAR1(PO,M1,NTOT,VAR,P,NB)
C FINDS THE VARIENCE OF NB OR IF NB IS SET EQUAL TO 0, THE SUM OF THE S
UARES OF THE NB SUBVALUES IS FOUND.
        DOUBLE PRECISION P(101),PO,M1,NTOT,VAR,NB
        ITOT=NTOT
        INITAL=NTOT-M1+1
        VAR=0.0
```

```
      IF(INITAL.EQ.0)VAR=VAR+P0*(M1-NTOT-NB)**2
      IF(INITAL.EQ.0)INITAL=1
      DO 10 NRESUP=INITAL,ITOT
10    VAR=VAR+P(NRESUP)*(M1-NTOT+NRESUP-NB)**2
      RETURN
      END
      SUBROUTINE VAR2(DER1,P,VAR,P0,NTOT)
      DOUBLE PRECISION P(101),DER1,VAR,P0,NTOT
      ITOT=NTOT
      VAR=P0*DER1**2
      DO 10 NRESUP=1,ITOT
10    VAR=VAR+P(NRESUP)*(NRESUP-DER1)**2
      RETURN
      END
$ENTRY
20    .5    15    17
/LOGOFF
```

POISSON MODEL

(INFINITE POPULATION)

PROGRAM LISTING

```
/LOGON    "NC"
/DO RUN.WATFIV
$JOB (L8NTJQQ)              KOVACS
      DOUBLE PRECISION P(101),PA(101),DER2(101),NB,NB1,PO,PO1,STOCK,STO
     1K1,NTOT,RO,M1,MO,ANS19,DER1,DER3,VAR,VARA,NBSQ,VARN,RESPSQ,VARS,M
     2,MN,VARM,U,SSQ,MNSQ,T
      READ,NTOT,RO,MO,M1
      ITOT=NTOT
      CALL PVAL(P,NTOT,MO,RO,PO)
      CALL NBSTK(NB,STOCK,NTOT,M1,MO,RO,PO)
      CALL PVAL(PA,NTOT+1,MO,RO,PO1)
      CALL NBSTK(NB1,STOCK1,NTOT+1,M1,MO,RO,PO1)
      PRINT142,NB-NB1
      CALL NINTEN(NTOT,M1,NB1,NB,PO1,PO,ANS19,MO,PA,RO)
142   FORMAT(' DELTA NB=',D20.14)
      PRINT400,ITOT,ITOT+1,ITOT,ANS19
400   FORMAT('ONB(',I3,') - NB(',I3,') / NB(',I3,') = ',D20.14)
      PRINT304,+(NB-NB1)/NB
304   FORMAT(' THE LEFT SIDE OF THE EQUATION GIVES ',D20.14)
      CALL SUB21(NTOT,P,DER1)
      PRINT310,DER1
310   FORMAT('OTHE EXPECTED NUMBER OF UNITS IN RESUPPLY IS',D20.14)
      PRINT320,DER1/(-RO)
320   FORMAT('OTHE RESULT OF FORMULA 21 IS ',D24.14)
      CALL SUB20(NTOT,RO,DER1,DER2)
      PRINT333
      PRINT330,0,DER1/(-RO)
```

7 2357

```
333   FORMAT('OANSWERS FOR FORMULA 20:')
330   FORMAT(' P(',I3,') GIVES ',D24.14)
      PRINT330,(J6,DER2(J6),J6=1,ITOT)
      CALL VAR1(PO,M1,NTOT,VAR,P,NB)
      CALL VAR1(PO1,M1,NTOT+1,VARA,PA,NB1)
      PRINT350,ITOT+1,VARA
350   FORMAT(' THE VARIANCE OF NB FOR NTOT=',I3,' IS ',D20.14)
      PRINT350,ITOT+1,VARA
      DER3=0.0
      CALL VAR1(PO,M1,NTOT,NBSQ,P,DER3)
      CALL SUB22(NTOT,M1,NB,DER1,DER3,RO,NBSQ)
      PRINT360,DER3
360   FORMAT(' THE ANSWER TO FORMULA 22 IS ',D20.14)
      PRINT352,NBSQ
      CALL VAR2(DER3,P,VARN,PO,NTOT)
352   FORMAT('OTHE EXPECTED VALUE OF THE DQUARE OF THE BACKORDER LEVELI
     1S:',D20.14)
      PRINT410,VARN
410   FORMAT('OTHE VARIANCE OF THE EXPECTED NUMBER OF UNITS IN RESUPPLY
     1IS:',D20.14)
      DER3=0.0
      CALL VAR2(DER3,P,RESPSQ,PO,NTOT)
      PRINT420,RESPSQ
420   FORMAT('OTHE EXPECTED VALUE OF THE SQUARE OF THE NUMBER OF UNITSI
     1N RESUPPLY IS:',D20.14)
      CALL SNAVE(NTOT,M1,P,PO,SN)
      PRINT430,SN
```

```
430   FORMAT('OTHE EXPECTED STOCK LEVEL IS',D21.14)
      CALL VAR3(NTOT,M1,P,PO,VARS,SN)
      PRINT440,VARS
440   FORMAT('OTHE VARIANCE OF THE STOCK LEVEL IS ',D20.14)
      U=0.0
      CALL VAR3(NTOT,M1,P,PO,SSQ,U)
425   FORMAT('OTHE EXPECTED VALUE OF THE SQUARE OF THE STOCK LEVEL IS:',
     1D20.14)
      PRINT425,SSQ
      CALL MNAVE(NTOT,M1,P,PO,MN)
      PRINT435,MN
435   FORMAT('OTHE EXPECTED NUMBER IN SERVICE IS ',D20.14)
      CALL VAR4(NTOT,M1,MN,P,VARM,PO)
      PRINT444,VARM
444   FORMAT('OTHE VARIANCE OF THE EXPECTED NUMBER IN SERVICE IS ',D20.
     14)
      CALL VAR4(NTOT,M1,U,P,MNSQ,PO)
      PRINT446,MNSQ
446   FORMAT('OTHE EXPECTED VALUE OF THE SQUARE OF THE NUMBER IN SERVIE
     1 IS:',D20.14)
      STOP
      END
      SUBROUTINE NINTEN(NTOT,M1,NB1,NB,PO1,PO,ANS19,MO,PA,RO)
      DOUBLE PRECISION NTOT,NBSQ,M1,NB,NB1,PO,PO1,ANS19,MO,PA(101),RO
      ANS19=0.0
      CALL VAR1(PO1,M1,NTOT+1,NBSQ,PA,ANS19)
      ANS19=1.-(1./(NTOT-M1+1))*(((MO*RO)*PO1/PO)-NBSQ/NB)
```

```
      RETURN
      END
      SUBROUTINE SUB20(NTOT,RO,DER1,DER2)
C FINDS THE RESULTS FOR FORMULA 20.
      DOUBLE PRECISION NTOT,RO,DER1,DER2(101)
      ITOT=NTOT
      DO 10 NRESUP=1,ITOT
10    DER2(NRESUP)=NRESUP/RO-DER1/RO
      PODER=DER1/(0.0-RO)
      RETURN                                          7  2359
      END
      SUBROUTINE SUB21(NTOT,P,DER1)
C FINDS THE EXPECTED NUMBER OF UNITS IN RESUPPLY.
      DOUBLE PRECISION NTOT,P(101),DER1
      DER1=0.0
      ITOT=NTOT
      DO 10 NRESUP=1,ITOT
10    DER1=NRESUP*P(NRESUP)+DER1
      RETURN
      END
      SUBROUTINE SUB22(NTOT,M1,NB,DER1,DER2,RO,NBSQ)
C FINDS THE RESULTS FOR FORMULA 22.
      DOUBLE PRECISION NTOT,M1,NB,DER1,DER2,RO,NBSQ
      DER2=NB/RO*((NTOT-M1)-DER1+NBSQ/NB)
      RETURN
      END
```

```
      SUBROUTINE VAR3(NTOT,M1,P,PO,VAR,SN)
      DOUBLE PRECISION NTOT,M1,P(101),PO,VAR,SN
      ITEST=NTOT-M1
      VAR=PO*(NTOT-M1-SN)**2
      DO 10 NRESUP=1,ITEST
10    VAR=VAR+P(NRESUP)*(NTOT-M1-NRESUP-SN)**2
      ITOT=NTOT
      INITAL=ITEST+1
      DO 30 NRESUP=INITAL,ITOT
30    VAR=VAR+P(NRESUP)*SN**2
      RETURN
      END
      SUBROUTINE SNAVE(NTOT,M1,P,PO,SN)
      DOUBLE PRECISION NTOT,M1,P(101),PO,SN
      ITEST=NTOT-M1
      SN=PO*(NTOT-M1)
      IF(ITEST.LT.1)GOTO 20
      DO 10 NRESUP=1,ITEST
      SN=SN+P(NRESUP)*(NTOT-M1-NRESUP)
10    CONTINUE
20    RETURN
      END
      SUBROUTINE MNAVE(NTOT,M1,P,PO,MN)
      DOUBLE PRECISION NTOT,M1,P(101),PO,MN
      INITAL=NTOT-M1+1
      MN=0
      IF(INITAL.LE.0)MN=PO*NTOT
```

```
      ITOT=NTOT
      IF(INITAL.LE.0)INITAL=1
      DO 10 NRESUP=INITAL,ITOT
10    MN=MN+(NTOT-NRESUP)*P(NRESUP)
      MN=MN+M1*P0
      ITEST=INITAL-1
      IF(ITEST.LT.1)GOTO 30
      DO 20 NRESUP=1,ITEST
20    MN=MN+M1*P(NRESUP)
30    RETURN
      END
      SUBROUTINE VAR4(NTOT,M1,MN,P,VAR,P0)
      DOUBLE PRECISION NTOT,M1,MN,P(101),VAR,P0
      INITAL=NTOT-M1
      ITEST=INITAL+1
      VAR=P0
      DO 20 NRESUP=1,ITEST
20    VAR=VAR+P(NRESUP)
      VAR=VAR*(M1-MN)**2
      IF(INITAL.EQ.0)VAR=VAR+P0*(NTOT-MN)**2
      IF(INITAL.EQ.0)INITAL=1
      ITOT=NTOT
      DO 10 NRESUP=INITAL,ITOT
10    VAR=VAR+P(NRESUP)*(NTOT-NRESUP-MN)**2
      RETURN
      END
```

7 2361

```
      SUBROUTINE FLOW(X,J,IX,IY)
C CHECKS FOR OVERFLOW AND UNDERFLOW --PRINTS WARNING MESSAGE IF ONE ISF
OUND.
      DOUBLE PRECISION X
      IF(X.LT.-174.9)IX=IX+1
      IF(X.GT.174.9)IY=IY+1
      IF(X.LT.-174.9)PRINT5,J
      IF(X.LT.-174.9)X=-174.9
5     FORMAT(' *****UNDERFLOW*****   J=',I3)
      IF(X.GT.174.9)PRINT10,J
10    FORMAT(' *****OVERFLOW*****   J=',I3)
      IF(X.GT.174.9)X=174.9
      RETURN
      END
      SUBROUTINE VAR1(PO,M1,NTOT,VAR,P,NB)
      DOUBLE PRECISION P(101),PO,M1,NTOT,VAR,NB
      ITOT=NTOT
      INITAL=NTOT-M1+1
      VAR=0.0
      IF(INITAL.EQ.0)VAR=VAR+PO*(M1-NTOT-NB)**2
      IF(INITAL.EQ.0)INITAL=1
      DO 10 NRESUP=INITAL,ITOT
10    VAR=VAR+P(NRESUP)*(M1-NTOT+NRESUP-NB)**2
      ITEST=NTOT-M1
      DO 20 NRESUP=1,ITEST
```

```
20     VAR=VAR+NB**2*P(NRESUP)
       VAR=VAR+PO*NB**2
       RETURN
       END
       SUBROUTINE VAR2(DER1,P,VAR,PO,NTOT)
       DOUBLE PRECISION P(101),DER1,VAR,PO,NTOT
       ITOT=NTOT
       VAR=PO*DER1**2
       DO 10 NRESUP=1,ITOT
10     VAR=VAR+P(NRESUP)*(NRESUP-DER1)**2
       RETURN
       END
       SUBROUTINE PVAL(P,NTOT,MO,RO,PO)
       DOUBLE PRECISION DLOG,DEXP,PO,MO,RO,X,C,B,A,P(101),NTOT
       ITEST=NTOT
       PO=DEXP(-MO*RO)
       X=PO
       DO 10 NRESUP=1,ITEST
       B=0.0
       DO 30 J=1,NRESUP
       A=J
30     B=B+DLOG(A)
       P(NRESUP)=(MO*RO)**NRESUP*DEXP(-MO*RO)/DEXP(B)
       X=X+P(NRESUP)
       PRINT20,NRESUP,P(NRESUP)
10     CONTINUE
```

```
20    FORMAT(' P(',I3,') = ',D20.14)
      PRINT40,X
40    FORMAT('0THE TOTAL OF THE POSSION PROBABILITIES IS',D20.14)
      RETURN
      END
      SUBROUTINE NBSTK(NB,STOCK,NTOT,M0,M1,R0,P0)
      DOUBLE PRECISION DLOG,DEXP,NTOT,M1,NB,STOCK,D,C,M0,R0,P0
      M=NTOT-M1+1
      ITEST=100
      NB=STOCK=0.0
      IF(M.LE.0)NB=P0
      IF(M.LE.0)M=1
      DO 50 NRESUP=M,ITEST
      C=0.0
      IF(NRESUP.LE.1)GOTO 65
      DO 60 J=2,NRESUP
      D=J
60    C=C+DLOG(D)
65    IF(NRESUP*DLOG(M0*R0)-C-M0*R0.LT.-100)GOTO111
50    NB=NB+(M1-NTOT+NRESUP)*DEXP(NRESUP*DLOG(M0*R0)-C)*DEXP(-M0*R0)
111   CONTINUE
      PRINT100,NB
100   FORMAT('0THERE ARE',D20.14,' BACKORDERS')
      RETURN
      END
$ENTRY
20    .2    15    17
/LOGOFF
```

CONSERVATIVE SINGLE SERVER
INVENTORY MODEL
(FINITE POPULATION)
PROGRAM LISTING

```
/LOGON  "NC"
/DO RUN.WATFIV
$JOB (LSNTJQQ)                    KOVACS
      DOUBLE PRECISION P(101),PA(101),DER2(101),NB,NB1,PO,PO1,Z,NTOT,RO
     1M1,MO,ANS36,DER1,DER3,VAR,VARA,NBSQ,VARN,RESFSQ,VARS,U,SSQ,MNSQ,T
     2ANS19
      Z=0.0
      PRINT5
      READ,K1,K2,K3,K4
5     FORMAT('1',5X,'** CONSERVATIVE SINGLE SERVER INVENTORY MODEL **')
      READ,MIN,MAX,RO,MO,M1
      PRINT10,MIN,MAX,RO,MO,M1
10    FORMAT('0MIN=',I4,4X,'MAX=',I4,4X,'RO=',D20.14,4X,'MO=',D20.14,4X
     1'M1=',D20.14)
      DO 1000 ITOT=MIN,MAX
      NTOT=ITOT
      IF(K4.EQ.1)PRINT150
      IX=IY=0
      IF(MO*RO.LE.1.0)CALL PVAL(P,NTOT,MO,RO,PO,K2)
      IF(MO*RO.GT.1.0)CALL PVAL1A(P,NTOT,MO,RO,PO,IX,IY,K2)
C IX AND IY ARE TOTALS FOR UNDER AND OVERFLOWS IN THE PROGRAM.
150   FORMAT('0P(N) IS THE PROBABILITY THAT N ITEMS ARE IN RESUPPLY AST
     1IME NEARS INFINITY.')
C T IS THE TOTAL OF THE PROBABILITIES -- SHOULD EQUAL 1.000.
      T=0.
      DO 100 I=1,ITOT
```

```
100    T=T+P(I)
       T=T+P0
       IF(K2.EQ.1)PRINT160,T
160    FORMAT('0THE TOTAL OF ALL THE P VALUES IS ',D20.14)
       CALL SUB17(NTOT,M1,P,K1)
       CALL SUB18(NTOT,M1,M0,R0,NB,K1)
       I6=NTOT-M1+1
       Z=0.0
       DO 300 I7=I6,ITOT
300    Z=Z+P(I7)
C ZIS THE TOTAL OF ALL THE PROBABILITIES FROM I=NTOT-M1+1 TO I=NTOT
       IF(K1.EQ.1)PRINT301,Z
301    FORMAT('0THE PROBABILITY FOR STOCKOUT IS',D20.14)
       CALL STOCK(M0,R0,M1,NTOT,K1)
       IX=IY=0
       K5=0
       IF(M0*R0.LE.1.)CALL PVAL(PA,NTOT+1,M0,R0,P01,K5)
       IF(M0*R0.GT.1.0)CALL PVAL1A(PA,NTOT+1,M0,R0,P01,IX,IY,K5)
       I79=ITOT+1
       PRINT220,IX,IY
220    FORMAT('0THERE WERE ',I3,' UNDERFLOWS AND ',I3,' OVERFLOWS')
       T=0.0
       DO 102 I=1,I79
102    T=T+PA(I)
       CALL SUB18(NTOT+1,M1,M0,R0,NB1,K5)
       CALL SUB36(M0,R0,P0,P01,ANS36)
       IF(K4.EQ.1)PRINT400,ITOT,ITOT+1,ITOT,ANS36
```

```
400    FORMAT('ONB(',I3,') -NB(',I3,') / NB(',I3,') = ',D20.14)
       IF(K4.EQ.1)PRINT304,+(NB-NB1)/NB
304    FORMAT('OTHE LEFT SIDE OF THE EQUATION GIVES ',D20.14)
       CALL SUB38(NTOT,P,DER1)
       IF(K3.EQ.1)PRINT310,DER1
310    FORMAT('OTHE EXPECTED NUMBER OF UNITS IN RESUPPLY IS',D20.14)
       IF(K4.EQ.1)PRINT320,-DER1/RO
320    FORMAT('OTHE RESULT OF FORMULA 38 IS ',D20.14)
       CALL SUB37(NTOT,RO,DER1,DER2)
       IF(K4.EQ.1)PRINT333
333    FORMAT('OTHE ANSWERS FOR FORMULA 37:')
       IF(K4.EQ.1)PRINT330,0,-DER1/RO
       IF(K4.EQ.1)PRINT330,(J5,DER2(J5),J5=1,ITOT)
330    FORMAT(' P(',I3,') GIVES ',D20.14)
       CALL VAR1(PO,M1,NTOT,VAR,P,NB)
       IF(K3.EQ.1)PRINT350,ITOT,VAR
350    FORMAT(' THE VARIENCE OF NB FOR NTOT=',I3,' IS ',D20.14)
       DER3=0.0
       CALL VAR1(PO,M1,NTOT,NBSQ,P,DER3)
       CALL SUB39(NTOT,M1,NB,DER1,DER3,RO,NBSQ)
       IF(K4.EQ.1)PRINT360,DER3
360    FORMAT('OTHE ANSWER TO FORMULA 39 IS ',D20.14)
       IF(K3.EQ.1)PRINT352,NBSQ
       CALL VAR2(DER1,P,VARN,PO,NTOT)
352    FORMAT('OTHE EXPECTED VALUE OF THE SQUARE OF THE BACKORDER LEVEL I
      1S:',D20.14)
       IF(K3.EQ.1)PRINT410,VARN
```

```
410   FORMAT('0THE VARIANCE OF THE EXPECTED NUMBER OF UNITS IN RESUPPLY
     1IS:',D20.14)
      DER3=0.
      CALL VAR2(DER3,P,RESPSQ,PO,NTOT)
      IF(K3.EQ.1)PRINT420,RESPSQ
420   FORMAT('0THE EXPECTED VALUE OF THE SPUARE OF THE NUMBER OF UNITSI
     1N RESUPPLY IS:',D20.14)
      CALL SNAVE(NTOT,M1,P,PO,SN)
      IF(K3.EQ.1)PRINT430,SN
430   FORMAT('0THE EXPECTED STOCK LEVEL IS ',D20.14)
      CALL VAR3(NTOT,M1,P,PO,VARS,SN)
      IF(K3.EQ.1)PRINT440,VARS
440   FORMAT('0THE VARIANCE OF THE STOCK LEVEL IS ',D20.14)
      U=0.
      CALL VAR3(NTOT,M1,P,PO,SSQ,U)
425   FORMAT('0THE EXPECTED VALUE OF THE SQUARE OF THE STOCK LEVEL IS:',
     1D20.14)
      IF(K3.EQ.1)PRINT425,SSQ
      CALL MNAVE(NTOT,M1,P,PO,MN)
      IF(K3.EQ.1)PRINT435,MN
435   FORMAT('0THE EXPECTED NUMBER IN SERVICE IS ',D20.14)
      CALL VAR4(NTOT,M1,MN,P,VARM,PO)
      IF(K3.EQ.1)PRINT444,VARM
444   FORMAT('0THE VARIANCE OF THE EXPECTED NUMBER IN SERVICE IS:',D20.
     14)
      CALL VAR4(NTOT,M1,U,P,MNSQ,PO)
      IF(K3.EQ.1)PRINT446,MNSQ
```

```
446    FORMAT('0THE EXPECTED VALUE OF THE SQUARE OF THE NUMBER IN SERVIE
      1 IS:',D20.14)
1000  CONTINUE
      STOP
      END
      SUBROUTINE PVAL(P,NTOT,MO,RO,PO,K2)
      DOUBLE PRECISION P(101),NTOT,MO,RO,PO,G,A,B,C,D,E
      DOUBLE PRECISION DLOG,DEXP
      ITEST=NTOT+1
      DO 5 JK=1,ITEST
      NRESUP=JK-1
      CALL GMORO(MO,RO,G)
      M=MO
      CALL FACT(M,C)
      A=-MO*DLOG(MO)+C
      B=NTOT*DLOG(MO*RO)
      A=A+B+G
      B=0.0
      I=NTOT-MO+1
      DO 10 M8=1,I
      J=M8-1
10    B=B+DEXP(J*DLOG(MO*RO))
      A=-(DLOG(DEXP(A)+B))
      IX=IY=0
      IF(NRESUP.EQ.0.0)CALL FLOW(A,NRESUP,IX,IY)
      IF(NRESUP.EQ.0.0)PO=DEXP(A)
      IF(NRESUP.EQ.0.0.AND.K2.EQ.1)PRINT101,NRESUP,PO
```

7 231

```
      IF(NRESUP.EQ.0.0)GOTO 5
      IF(NRESUP.LE.NTOT-MO)P(NRESUP)=DEXP((NRESUP*DLOG(MO*RO))+A)
      D=(NTOT-MO)*DLOG(MO)
      M9=NTOT-NRESUP
      G=0.0
      CALL FACT(M9,G)
      E=NRESUP*DLOG(RO)
      IF(NRESUP.GT.NTOT-MO)P(NRESUP)=DEXP(C-G+D+E+A)
      IF(K2.EQ.1)PRINT101,NRESUP,P(NRESUP)
5     CONTINUE
101   FORMAT(' P(',I3,') = ',D20.14)
      RETURN
      END
      SUBROUTINE SUB17(NTOT,M1,P,K1)
      DOUBLE PRECISION NTOT,M1,P(101),NB,X
      NB=0.0
      M=M1
      DO 10 J=1,M
      X=J-1
      I=NTOT-X
      NB=NB+(M1-X)*P(I)
10    CONTINUE
      IF(K1.EQ.1)PRINT40,NB
40    FORMAT('0FORMULA 17 GIVES',D20.14,' BACKORDERS')
      RETURN
      END
```

```
      SUBROUTINE SUB36(MO,RO,PO,PO1,ANS36)
      DOUBLE PRECISION MO,RO,PO,PO1,ANS36
      ANS36=1.0-MO*RO*PO1/PO
      RETURN
      END
      SUBROUTINE SUB37(NTOT,RO,DER1,DER2)
      DOUBLE PRECISION NTOT,RO,DER1,DER2(101)
      ITOT=NTOT
      DO 10 NRESUP=1,ITOT
10    DER2(NRESUP)=NRESUP/RO-DER1/RO
      PODER=DER1/(0.0-RO)
      RETURN
      END
      SUBROUTINE SUB38(NTOT,P,DER1)
      DOUBLE PRECISION P(101),NTOT,DER1
      DER1=0.0
      ITOT=NTOT
      DO 10 NRESUP=1,ITOT
10    DER1=NRESUP*P(NRESUP)+DER1
      RETURN
      END
      SUBROUTINE SUB39(NTOT,M1,NB,DER1,DER2,RO,NBSQ)
      DOUBLE PRECISION NTOT,M1,NB,DER1,DER2,RO,NBSQ
      DER2=NB/RO*((NTOT-M1)-DER1+NBSQ/NB)
      RETURN
      END
```

```
      SUBROUTINE VAR3(NTOT,M1,P,PO,VAR,SN)
      DOUBLE PRECISION NTOT,M1,P(101),PO,VAR,SN
      ITEST=NTOT-M1
      VAR=PO*(NTOT-M1-SN)**2
      DO 10 NRESUP=1,ITEST
10    VAR=VAR+P(NRESUP)*(N-M1-NRESUP-SN)**2
      ITOT=NTOT
      INITAL=ITEST+1
      DO 30 NRESUP=INITAL,ITOT
30    VAR=VAR+P(NRESUP)*SN**2
      RETURN
      END
      SUBROUTINE SNAVE(NTOT,M1,P,PO,SN)
      DOUBLE PRECISION NTOT,M1,P(101),PO,SN
      ITEST=NTOT-M1
      SN=PO*(NTOT-M1)
      IF(ITEST.LT.1)GOTO 20
      DO 10 NRESUP=1,ITEST
      SN=SN+P(NRESUP)*(NTOT-M1-NRESUP)
10    CONTINUE
20    RETURN
      END
      SUBROUTINE MNAVE(NTOT,M1,P,PO,MN)
      DOUBLE PRECISION NTOT,M1,P(101),PO,MN
      INITAL=NTOT-M1+1
      MN=0
      IF(INITAL.EQ.0)MN=PO*(NTOT)
```

```
      ITOT=NTOT
      IF(INITAL.EQ.0)INITAL=1
      DO 10 NRESUP=INITAL,ITOT
10    MN=MN+(NTOT-NRESUP)*P(NRESUP)
      MN=MN+M1*P0
      ITEST=INITAL-1
      IF(ITEST.LT.1)GOTO 30
      DO 20 NRESUP=1,ITEST
20    MN=MN+M1*P(NRESUP)
30    RETURN
      END
      SUBROUTINE VAR4(NTOT,M1,MN,P,VAR,P0)
      DOUBLE PRECISION NTOT,M1,MN,P(101),VAR,P0
      INITAL=NTOT-M1+1
      ITEST=INITAL-1
      VAR=P0
      DO 20 NRESUP=1,ITEST
20    VAR=VAR+P(NRESUP)
      VAR=VAR*(M1-MN)**2
      IF(INITAL.EQ.0)VAR=VAR+P0*(NTOT-MN)**2
      IF(INITAL.EQ.0)INITAL=1
      ITOT=NTOT
      DO 10 NRESUP=INITAL,ITOT
10    VAR=VAR+P(NRESUP)*(NTOT-NRESUP-MN)**2
      RETURN
      END
```

```
      SUBROUTINE FLOW(X,J,IX,IY)
      DOUBLE PRECISION X
      IF(X.LT.-174.9)IX=IX+1
      IF(X.GT.174.9)IY=IY+1
      IF(X.LT.-174.9)PRINT5,J
5     FORMAT(' *****UNDERFLOW*****    J=',I3)
      IF(X.GT.174.9)PRINT10,J
10    FORMAT(' *****OVERFLOW*****   J=',I3)
      IF(X.LT.-174.9)X=-174.9
      IF(X.GT.174.9)X=174.9
      RETURN
      END
      SUBROUTINE VAR1(PO,M1,NTOT,VAR,P,NB)
      DOUBLE PRECISION P(101),PO,M1,NTOT,VAR,NB
      ITOT=NTOT
      INITAL=NTOT-M1+1
      VAR=0.0
      IF(INITAL.EQ.0)VAR=VAR+PO*(M1-NTOT-NB)**2
      IF(INITAL.EQ.0)INITAL=1
      DO 10 NRESUP=INITAL,ITOT
10    VAR=VAR+P(NRESUP)*(M1-NTOT+NRESUP-NB)**2
      ITEST=NTOT-M1
      DO 20 NRESUP=1,ITEST
20    VAR=VAR+NB**2*P(NRESUP)
      VAR=VAR+PO*NB**2
      RETURN
      END
```

```
      SUBROUTINE VAR2(DER1,P,VAR,PO,NTOT)
      DOUBLE PRECISION P(101),DER1,VAR,PO,NTOT
      ITOT=NTOT
      VAR=PO*DER1**2
      DO 10 NRESUP=1,ITOT
10    VAR=VAR+P(NRESUP)*(NRESUP-DER1)**2
      RETURN
      END
      SUBROUTINE SUB19(NB,MO,RO,NTOT,M1,ANS19)
      DOUBLE PRECISION MO,RO,NB,M1,NTOT,ANS19,G,B,A
      DOUBLE PRECISION DLOG,DEXP
      CALL GMORO(MO-1.0,RO,G)
      G=DEXP(G)
      M=MO
      IF(MO-1.0.LE.0.0)G=0.0
      CALL FACT(M,A)
      B=MN-NB-1.0/RO
      IF(B.LT.1E-12.AND.B.GT.0.0)B=0.0
      A=((1.0-MO*RO)*DEXP(-MO*DLOG(MO)+A)*G*B)+(M1-(MO*RO)*(M1-NB))*DEX
     1((1.0-MO)*DLOG(MO*RO))+((1.0-DEXP((M1-MO)*DLOG(MO*RO)))*DEXP(-M1*
     2LOG(MO*RO))/(1.0-MO*RO))-(M1-MO)*DEXP(-MO*DLOG(MO*RO))
      ANS19=(DLOG(A)-DLOG(NB))/(-DLOG(MO*RO))
      RETURN
      END
      SUBROUTINE FACT(M,A)
      DOUBLE PRECISION A,B
      DOUBLE PRECISION DLOG
```

```
      A=0.0
      IF(M.LE.1)GOTO 20
      DO 10 J=2,M
      B=J
10    A=A+DLOG(B)
20    RETURN
      END
      SUBROUTINE GMORO(MO,RO,A)
      DOUBLE PRECISION MO,RO,A,B,DLOG,DEXP
      INTEGER X
      M=MO
      A=0.0
      DO 10 J=1,M
      X=J-1
      B=0.0
      CALL FACT(X,B)
10    A=A+DEXP(-X*DLOG(RO)-B)
      A=DLOG(A)
      RETURN
      END
      SUBROUTINE SUB21(MO,RO,NB,M1)
      DOUBLE PRECISION MO,RO,NB,M1,G,A,B,ANS
      DOUBLE PRECISION DLOG,DEXP
      IF(MO.NE.M1)GOTO 20
      CALL GMORO(MO-1.0,RO,G)
      G=DEXP(G)
      IF(MO-1.0.LE.0)G=0.0
      M=MO
```

```
      CALL FACT(M,A)
      B=M1-NB-1.0/RO
      IF(B.LT.1E-12.AND.B.GT.0.0)B=0.0
      A=((MO-(MO*RO)*(MO-NB))*DEXP((1.0-MO)*DLOG(MO*RO)))+(DEXP(-MO*DLG
     1(MO)+A)*G*(1.0-MO*RO)*B)
      ANS=(DLOG(NB)-DLOG(A))/(DLOG(MO*RO))
      PRINT10,ANS
10    FORMAT('OFORMULA 21 GIVES ',D20.14)
20    RETURN
      END
      SUBROUTINE PVAL1A(P,NTOT,MO,RO,PO,IX,IY,K2)
      DOUBLE PRECISION P(101),NTOT,MO,RO,PO,G,A,B,C,D,E,F
      DOUBLE PRECISION DLOG,DEXP
      ITEST=NTOT+1
      DO 5 JK=1,ITEST
      NRESUP=JK-1
      CALL GMORO(MO,RO,G)
      M=MO
      CALL FACT(M,C)
      A=-MO*DLOG(MO)+C+G
      B=0.0
      I=NTOT-MO+1
      IF(I.LE.0)GOTO 20
      DO 10 MS=1,I
      J=MS-1
10    B=B+DEXP((J-NTOT)*DLOG(MO*RO))
```

```
20      A=-(DLOG(DEXP(A)+B))
        IF(NRESUP.EQ.0.0)CALL FLOW(A,NRESUP,IX,IY)
        IF(NRESUP.EQ.0)PO=DEXP(-NTOT*DLOG(MO*RO)+A)
        IF(NRESUP.EQ.0.AND.K2.EQ.1)PRINT101,NRESUP,PO
        IF(NRESUP.EQ.0.0)GOTO 5
        IF(NRESUP.LE.NTOT-MO)P(NRESUP)=DEXP((NRESUP-NTOT)*DLOG(MO*RO)+A)
        D=-MO*DLOG(MO)
        M9=NTOT-NRESUP
        F=0.0
        CALL FACT(M9,F)
        E=(NRESUP-NTOT)*DLOG(RO)
        IF(NRESUP.GT.NTOT-MO)P(NRESUP)=DEXP(C-F+D+E+A)
        IF(K2.EQ.1)PRINT101,NRESUP,P(NRESUP)
5       CONTINUE
101     FORMAT(' P(',I3,') = ',D20.14)
        RETURN
        END
        SUBROUTINE SUB18(NTOT,M1,MO,RO,NB,K1)
        DOUBLE PRECISION NTOT,MO,M1,RO,NB,Q,GAMMA,BETA,A,B,C,D,E,F,G,G1,M
       1R,S,T,U,DLOG,DEXP
        M=MO
        G=0.0
        IF(MO.LE.1)GOTO 5
        CALL GMORO(MO-1,RO,G)
        G=DEXP(G)
```

```
5     CALL FACT(M,U)
      A=(M1-1.0/RO)*DEXP(-MO*DLOG(MO)+U)*G
      B=DEXP(DLOG(M1)+(1.0-MO)*DLOG(MO*RO))
      GAMMA=M1-MO
      BETA=1.0-MO*RO
      C=0.0
      M9=GAMMA-1
      IF(GAMMA.LT.2.0)GOTO 20
      DO 10 J=1,M9
      K=J-1
      CALL FACT(M9+2,D)
      CALL FACT(K,E)
      J5=GAMMA-K-1
      CALL FACT(J5,F)
      H=K+2
10    C=C+(-1)**K*BETA**K*DEXP(D-E-F-DLOG(H))
20    R=0.0
      IF(BETA.EQ.0.0.OR.GAMMA.EQ.0.0)GOTO 25
      R=(-1)**(M9+2)*BETA**M9*GAMMA
25    IF(M1.EQ.1.0)Q=C+R
      IF(M1.NE.1.0)Q=DEXP((1.0-M1)*DLOG(MO*RO))*(C+R)
      J26=NTOT-MO+1
      S=0.0
      DO 30 J27=1,J26
      J=J27-1
30    S=S+(DEXP((J-NTOT)*DLOG(MO*RO)))
      CALL GMORO(MO,RO,G1)
      T=DEXP(-MO*DLOG(MO)+U+G1)
```

```
      NB=DEXP(DLOG(A+B+Q)-DLOG(S+T))
      IF(K1.EQ.1)PRINT50,NB
      IF(K1.EQ.1)PRINT70,+DEXP(DLOG(A+B+Q)-DLOG(T))
50    FORMAT('OFORMULA 18 GIVES',D20.14,' BACKORDERS')
70    FORMAT('OFOR N=INFINITY, NB =',D20.14)
      RETURN
      END
      SUBROUTINE STOCK(MO,RO,M1,NTOT,K1)
      DOUBLE PRECISION MO,RO,M1,NTOT,G,A,B,C,D,E,F,STOK
      DOUBLE PRECISION DLOG,DEXP
      M=MO
      CALL FACT(M,A)
      CALL GMORO(MO,RO,G)
      B=DEXP(-MO*DLOG(MO)+A+G)
      C=0.0
      I=M1-MO
      IF(I.LT.1)GOTO 20
      DO 10 N=1,I
      J=N-1
      IF(M1-J.EQ.1.0)C=C+1
      IF(M1-J.EQ.1.0)GOTO 10
      C=C+DEXP((J-M1+1)*DLOG(MO*RO))
10    CONTINUE
20    D=0.0
      IJ=NTOT-MO+1
      DO 30 N=1,IJ
      J=N-1
```

```
30      D=D+DEXP(J*DLOG(MO*RO))
        E=DEXP(NTOT*DLOG(MO*RO))
        F=DEXP(-NTOT*DLOG(MO*RO))
        IF(MO*RO.LE.1)STOK=DEXP(DLOG(B+C)-DLOG(D*F+B))
        IF(MO*RO.GT.1.0)STOK=DEXP(DLOG(E*(B+C))-DLOG(D+B*E))
        IF(K1.EQ.1)PRINT50,STOK
50      FORMAT('0THE STOCKOUT PROBABILITY IS',D20.14)
        RETURN
        END
$ENTRY
1       1       1       1
10      15      .5      11      13
/LOGOFF
```

PROBLEM 1

PROGRAM LISTING

```
DCUBLE PRECISION N,NBO,U,L
DLUBLE PRECISION A,B,C,D,RO1,RO2,RO3,RO4,ZO,RO,A21,M1,MO,
          FN1,FN2,FN3,P(101),PO,Q,Q1,Q2,Q3,Q4,Q5,Q6,Q7,Q8
       INTEGER DN1,DN2
       READ RO,M1,MO,NBO,NMAX,A21
       NMIN=1
       N=NMAX/2
       IF(N.LT.1)N=1
       A,B,C,D,RO1,RO2,RO3,RO4=0.0
5      IF(N.EQ.1)GOTO 10
       CALL FIND(N-1,RO1,M1,MO,NBO,A21)
       A=(N-1)*(1+RO/RO1)
10     CALL FIND(N,RO2,M1,MO,NBO,A21)
       B=N*(1+RO/RO2)
       IF(N.EQ.NMAX)GOTO 30
       CALL FIND(N+1,RO3,M1,MO,NBO,A21)
       C=(N+1)*(1+RO/RO3)
30     IF(NMAX-1.0.EQ.N)GOTO 40
       CALL FIND(N+2,RO4,M1,MO,NBO,A21)
       D=(N+2)*(1+RO/RO4)
40     FN1=C-B
       FN2=B-A
       FN3=D-C
       Q=N
       Q1=B
       Q2=RO2
       Q3=N-1
       Q4=A
       Q5=RO1
       Q6=N+1
       Q7=C
       Q8=RO3
       IF(FN1.GE.0.0.AND.FN2.LE.0.0)GOTO 100
       IF(N.EQ.NMAX.AND.FN2.LE.0.0)GOTO 100
       IF(N.EQ.1.0.AND.FN1.GE.0.0)GOTO 100
       Q=N+1
       Q1=C
       Q2=RO3
       Q3=N
       Q4=B
       Q5=RO2
       Q6=N+2
       Q7=D
       Q8=RO4
       IF(FN3.GE.0.0.AND.FN1.LE.0.0)GOTO 100
       IF(FN1.LE.0.0.AND.NMAX-1.0.EQ.N)GOTO 100
       Q=N+2
       Q1=D
       Q2=RO4
       Q3=N+1
       Q4=C
       Q5=RO3
       Q6,Q7,Q8=0.0
```

```
      IF(NMAX-2.0.EQ.N.AND.FN3.LE.0.0)GOTO 100
      Q=1
      Q1=A
      Q2=RO1
      Q3,Q4,Q5=0.0
      Q6=2
      Q7=B
      Q8=RO2
      IF(N.EQ.2.0.AND.FN2.GE.0.0)GOTO 100
      IF(FN1.EQ.0)PRINT,'ERROR #1'
      IF(FN2.EQ.0.0)PRINT,'ERROR #2'
      DN1=1/(1-(FN3/FN1))
      DN2=1/(1-(FN1/FN2))
      IF(DN1.LT.DN2.AND.-DN1.GT.DN2)N=N+DN1
      IF(DN1.LT.DN2.AND.-DN1.LE.DN2)N=N+DN2
      IF(DN2.LE.DN1.AND.-DN2.GT.DN1)N=N+DN2
      IF(DN2.LE.DN1.AND.-DN2.LE.DN1)N=N+DN1
      IF(N.LE.0)N=1
      IF(N.GT.NMAX)N=NMAX
      PRINT,N,B,RO2
      GOTO 5
100   PRINT200,Q3,Q4,Q5
      PRINT200,Q,Q1,Q2
      PRINT200,Q6,Q7,Q8
      U=(1+(Q8-Q2)/Q2)/((Q*(Q8-Q2)/Q2)-1)
      L=(1-(Q2-Q5)/Q2)/((Q*(Q2-Q5)/Q2)-1)
      PRINT210,L,U
      PRINT220,L*Q2,U*Q2
210   FORMAT(1X,D20.14,'<(RO/RO1)<',D20.14)
220   FORMAT(1X,D20.14,'<RO<',D20.14)
200   FORMAT('ON=',D20.14,5X,'Z=',D20.14,/,' RO1=',D20.14)
      PRINT900
900   FORMAT('1')
      PAUSE
      STOP
      END
      SUBROUTINE SUB21(NTOT,P,DER1)
C FINDS THE EXPECTED NUMBER OF UNITS IN RESUPPLY.
      D.P. NTOT,P(101),DER1
      DER1=0.0
      ITOT=NTOT
      DO 10 NRESUP=1,ITOT
10    DER1=NRESUP*P(NRESUP)+DER1
      RETURN
      END
      SUBROUTINE NINE(NTOT,MO,P,PO,RO,NB,M1)
C FINDS THE PROBABILITIES FOR J=0 TO J=NTOT-MO (USES NATURAL
      LOGSTO ALLOW FOR LARGER NTOT).
      DOUBLE PRECISION P(101),NTOT,PO,MO,RO,A,C,B,D,E,F,G,H,Z,Y,NB,M1
      ITEST=NTOT-MO+1
      I11=ITEST
```

```
      IF(I11.LT.1)I11=1
      DO 10 J1=1,ITEST
      J=J1-1
      Y,Z=0.0
      IF(ITEST.LT.1)GOTO 21
      DO 20 N1=1,ITEST
      NRESUP=N1-1
      A=(NRESUP-J)*LOG(MO*RO)
      C,D=0.0
      IF(J.LE.1)GOTO 35
      DO 30 K=2,J
      B=K
30    C=C+LOG(B)
35    IF(NRESUP.LE.1)GOTO 45
      DO 40 K=2,NRESUP
      B=K
40    D=D+LOG(B)
45    E=A+C-D
      CALL FLOW(E,J,IX,IY)
      IF (E.LT.-174.996)GOTO 20
      Z=Z+EXP(E)
20    CONTINUE
21    IF(ITEST.LE.1)ITEST=1
      ITOT=NTOT+1
      ITEST=ITEST+1
      DO 50 N1=ITEST,ITOT
      NRESUP=N1-1
      C,D=0.0
      M=MO
      IF(M.LE.1)GOTO 65
      DO 60 K=2,M
      B=K
60    C=C+LOG(B)
65    IF(J.LE.1)GOTO 75
      DO 70 K=2,J
      B=K
70    D=D+LOG(B)
75    A,F=0.0
      IF(NRESUP.LE.1)GOTO 85
      DO 80 K=2,NRESUP
      B=K
80    A=A+LOG(B)
85    N=NTOT-NRESUP
      IF(N.LE.1)GOTO 95
      DO 90 K=2,N
      B=K
90    F=F+LOG(B)
95    G=(NTOT-MO-J)*LOG(MO)
      H=(NRESUP-J)*LOG(RO)
      E=C+D+G+H-A-F
      CALL FLOW(E,J,IX,IY)
      IF(E.LT.-174.996)GOTO 50
      Y=Y+EXP(E)
```

```
50      CONTINUE
        IF(J.EQ.0)PO=1.0/(Y+Z)
        IF(J.EQ.0)GOTO 10
        P(J)=1./(Y+Z)
10      CONTINUE
        CALL TEN(NTOT,MO,P,RO)
        CALL SIXTEN(NTOT,M1,P,NB,PO)
        RETURN
        END
        SUBROUTINE TEN(NTOT,MO,P,RO)
C FINDS THE PROBABILITIES FOR J=NTOT-MO+1 TO J=NTOT (USES NATURALLOGS
        TO ALLOW FOR LARGER NTOT).
        D.P.P(101),NTOT,PO,MO,RO,A,B,C,D,E,F,G,H,Z,Y
        INITAL=NTOT-MO+1
        IF(INITAL.LT.1)INITAL=1
        ITOT=NTOT
        DO 10 J=INITAL,ITOT
        Y,Z=0.0
        DO 20 N1=1,INITAL
        NRESUP=N1-1
        A,C,H,G=0.0
        IF(J.LE.1)GOTO 105
        DO 100 K=2,J
        B=K
100     G=G+LOG(B)
105     N=NTOT-J
        IF(N.LE.1)GOTO 115
        DO 110 K=2,N
        B=K
110     H=H+LOG(B)
115     IF(NRESUP.LE.1)GOTO 35
        DO 30 K=2,NRESUP
        B=K
30      A=A+LOG(B)
35      N=MO
        IF(N.LE.1)GOTO 45
        DO 40 K=2,N
        B=K
40      C=C+LOG(B)
45      D=(NRESUP-J)*LOG(RO)
        E=(NRESUP-(NTOT-MO))*LOG(MO)
        F=D+E+G+H-A-C
        CALL FLOW(F,J,IX,IY)
        IF (F.LT.-174.996)GOTO 20
        Y=Y+EXP(F)
20      CONTINUE
        DO 50 NRESUP=INITAL,ITOT
        A,C,D,E,F=0.0
        IF(J.LE.1)GOTO65
        DO 60 K=2,J
        B=K
60      A=A+LOG(B)
```

```
65     I=NTOT-J
       IF(I.LE.1)GOTO 75
       DO 70 K=2,I
       B=K
70     C=C+LOG(B)
75     I=NTOT-NRESUP
       IF(I.LE.1)GOTO 85
       DO 80 K=2,I
       B=K
80     E=E+LOG(B)
85     IF(NRESUP.LE.1)GOTO 95
       DO 90 K=2,NRESUP
       B=K
90     F=F+LOG(B)
95     D=(NRESUP-J)*LOG(RO)
       B=D+A+C-E-F
       CALL FLOW(B,J,IX,IY)
       IF(B.LT.-174.996)GOTO 50
       Z=Z+EXP(B)
50     CONTINUE
       P(J)=1.0/(Y+Z)
10     CONTINUE
       RETURN
       END
       SUBROUTINE SIXTEN(NTOT,M1,P,NB,PO)
C FINDS THE EXPECTED NUMBER OF BACKORDERS.
       DOUBLE PRECISION NTOT,M1,P(101),NB,PO,B
       ITOT=NTOT
       NB=0.
       INITAL=NTOT-M1+1
       IF(INITAL.LE.0)NB=PO
       IF(INITAL.LE.0)INITAL=1
       DO 10 NRESUP=INITAL,ITOT
       B=(M1-(NTOT-NRESUP))*P(NRESUP)
10     NB=NB+B
       RETURN
       END
       SUBROUTINE FLOW(X,J,IX,IY)
C CHECKS FOR OVERFLOW AND UNDERFLOW --PRINTS WARNING MESSAGE IF
       ONE IS FOUND.
       D.P.X
       IF(X.LT.-174.996)IX=IX+1
       IF(X.GT.174.996)IY=IY+1
       IF(X.LT.-174.996)PRINT5,J
5      FORMAT(' *****UNDERFLOW*****  J=',I3)
       IF(X.GT.174.996)PRINT10,J
10     FORMAT(' *****OVERFLOW*****  J=',I3)
       IF(X.GT.174.996)X=174.99
C SETS X EPUAL TO MAXIMUM VALUE WITH NO OVERFLOW IF AN OVERFLOW
       IS FOUND (LN(175)>10**76).
       RETURN
       END
       SUBROUTINE NBSTK(NB,NTOT,MO,M1,RO)
```

```
      D.P. NTOT,M1,NB,STOCK,D,C,MO,RO,PO
      M=NTOT-M1+1
      ITEST=100
      NB=0.0
      IF(M.LE.0)NB=EXP(-MO*RO)
      IF(M.LE.0)M=1
      DO 50 NRESUP=M,ITEST
      C=0.0
      IF(NRESUP.LE.1)GOTO 65
      DO 60 J=2,NRESUP
      D=J
60    C=C+LOG(D)
65    IF(NRESUP*LOG(MO*RO)-C-MO*RO.LT.-100)GOTO111
50    NB=NB+(M1-NTOT+NRESUP)*EXP(NRESUP*LOG(MO*RO)-C)*EXP(-MO*RO)
111   CONTINUE
      RETURN
      END
      SUBROUTINE FIND(NTOT,RO1,M1,MO,NB2A,A)
      DOUBLE PRECISION RO,RO1,NTOT,COUNT,MO,PB(101),POB,NB2,
      PC(101),POC,NB3,M1,DER5,DER1C,DRO,EPSILN,A,NB2A
      RO,RO1=1.0
1032  IF(A.NE.0.0)CALL NINE(NTOT-1,MO,PB,POB,RO1,NB2,M1)
      IF(A.NE.0.0)CALL NINE(NTOT,MO,PC,POC,RO1,NB3,M1)
      IF(A.NE.0.0)CALL SUB21(NTOT,PC,DER1C)
      IF(A.EQ.0.0)CALL NBSTK(NB3,NTOT,MO,M1,RO1)
      IF(A.EQ.0.0)CALL NBSTK(NB2,NTOT-1,MO,M1,RO1)
      IF(A.EQ.0.0)DER1C=MO*RO1
      COUNT=COUNT+1
      DER5=(-DER1C/RO1)*(NB3-NB2)
      DRO=(NB2A-NB3)/DER5
      RO1=RO1+DRO
      EPSILN=DRO/RO1
      IF(EPSILN.LE.2E-14.AND.EPSILN.GE.-2E-14)GOTO 1031
      IF(COUNT.GE.100)GOTO 1031
      GOTO 1032
1031  RETURN
      END
```

PROBLEM 2

PROGRAM LISTING

```
/LOGON "NC"
/DO RUN.WATFIV
$JOB (L8NTJQQ)                          KOVACS
      DOUBLE PRECISION N,ROP1,ROP2,ROA,ROB,ROC,DRO1P,DRO2P,A1,A3
      DOUBLE PRECISION A,B,C,D,RO1,RO2,RO3,RO4,ZO,RO,A21,M1,MO,FN1,FN2,
     1FN3,P(101),PO,Q,Q1,Q2,Q3,Q4,Q5,Q6,Q7,Q8
      INTEGER DN1,DN2
      READ(5,*) RO,MO,ZO
      DO 1000 M31=1,2
      M1=MO
      A21=M31-1.0
      PRINT,RO,MO,ZO,A21
      NMIN=1
      NMAX=ZO
      IF(NMAX.EQ.ZO)NMAX=ZO-1.0
      I=ZO/2
      IF(A21.EQ.0.0)N=I
      IF(N.LT.1)N=1
      A=B=C=D=RO1=RO2=RO3=RO4=0.0
5     IF(N.EQ.1)GOTO 10
      RO1=RO*(N-1)/(ZO-(N-1))
      IF(A21.EQ.0.0)CALL NBSTK(A,N-1,MO,M1,RO1)
      IF(A21.NE.0.0)CALL NINE(N-1,MO,P,PO,RO1,A,M1)
10    RO2=RO*N/(ZO-N)
      IF(A21.EQ.0.0)CALL NBSTK(B,N,MO,M1,RO2)
      IF(A21.NE.0.0)CALL NINE(N,MO,P,PO,RO2,B,M1)
```

```
      IF(N.EQ.NMAX)GOTO 30
      RO3=RO*(N+1)/(ZO-(N+1))
      IF(A21.EQ.0.0)CALL NBSTK(C,N+1,MO,M1,RO3)
      IF(A21.NE.0.0)CALL NINE(N+1,MO,P,PO,RO3,C,M1)
30    IF(NMAX-1.0.EQ.N)GOTO 40
      RO4=RO*(N+2)/(ZO-(N+2))
      IF(A21.EQ.0.0)CALL NBSTK(D,N+2,MO,M1,RO4)
      IF(A21.NE.0.0)CALL NINE(N+2,MO,P,PO,RO4,D,M1)
40    FN1=C-B
      FN2=B-A
      FN3=D-C
      Q=N
      Q1=B
      Q2=RO2
      Q3=N-1
      Q4=A
      Q5=RO1
      Q6=N+1
      Q7=C
      Q8=RO3
      IF(FN1.GE.0.0.AND.FN2.LE.0.0)GOTO 100
      IF(N.EQ.NMAX.AND.FN2.LE.0.0)GOTO 100
      IF(N.EQ.1.0.AND.FN1.GE.0.0)GOTO 100
      Q=N+1
      Q1=C
      Q2=RO3
      Q3=N
```

```
Q4=B
Q5=RO2
Q6=N+2
Q7=D
Q8=RO4
IF(FN3.GE.0.0.AND.FN1.LE.0.0)GOTO 100
IF(FN1.LE.0.0.AND.NMAX-1.0.EQ.N)GOTO 100
Q=N+2
Q1=D
Q2=RO4
Q3=N+1
Q4=C
Q5=RO3
Q6=Q7=Q8=0.0
IF(NMAX-2.0.EQ.N.AND.FN3.LE.0.0)GOTO 100
Q=1
Q1=A
Q2=RO1
Q3=Q4=Q5=0.0
Q6=2
Q7=B
Q8=RO2
IF(N.EQ.2.0.AND.FN2.GE.0.0)GOTO 100
IF(FN1.EQ.0)PRINT,'ERROR #1'
IF(FN2.EQ.0.0)PRINT,'ERROR #2'
DN1=1/(1-(FN3/FN1))
DN2=1/(1-(FN1/FN2))
```

```
      IF(IABS(DN1).GE.IABS(DN2))ND=DN2
      IF(IABS(DN2).GT.IABS(DN1))ND=DN1
      IF(ND.NE.0)GOTO 98
      IF(FN1.GT.0.0)ND=-1
      IF(FN1.LT.0.0)ND=2
98    IF(ND.EQ.1)ND=2
      IF(FN1.LT.0.0.AND.ND.LT.0)ND=2
      N=N+ND
      IF(N.LE.0)N=1
      IF(N.GT.NMAX)N=NMAX
      PRINT,N,B,RO2
      GOTO 5
100   PRINT200,Q3,Q4,Q5
      PRINT200,Q,Q1,Q2
      PRINT200,Q6,Q7,Q8
200   FORMAT(' CN=',D20.14,5X,'NB=',D20.14,/,' RO1=',D20.14)
      N=Q
      ROP1=RO
      COUNT=0.0
300   ROA=(N+1)*ROP1/(Z0-N-1)
      ROB=N*ROP1/(Z0-N)
      COUNT=COUNT+1
      CALL HELP(N+1,M0,ROA,A1,M1,A,A21)
      CALL HELP(N,M0,ROB,A3,M1,B,A21)
      IF((A*ROA-B*ROB).EQ.0.0)PRINT301
      IF((A*ROA-B*ROB).EQ.0.0.AND.(A1-A3).LE.2E-14)Q=0.0
      IF((A1-A3).GE.-2E-14.AND.Q.EQ.0.0)ROP1=0.0
      IF(ROP1.EQ.0.0)GOTO 310
```

```
301     FORMAT('OERROR #1')
        IF(COUNT.EQ.1.0)GOTO 309
        Q2=-(A1-A3)/((A*ROA-B*ROB)/ROP1)-DRO1P
        IF(Q2.LE.1E-14.AND.COUNT.GE.50.0)ROP1=0.0
        IF(ROP1.EQ.0.0)GOTO310
309     DRO1P=-(A1-A3)/((A*ROA-B*ROB)/ROP1)
        IF(COUNT.GT.2)GOTO 307
        IF(DRO1P.LT.0.0)COUNT=3.0
        IF(COUNT.EQ.3.0)GOTO 307
        IF(COUNT.EQ.1.0)DRO1P=-DRO1P
        IF(COUNT.EQ.1.0)Q7=A1-A3
        IF(COUNT.EQ.1.0)GOTO 307
        IF((A1-A3).GT.0.0.AND.Q7.LT.0.0)GOTO 307
        IF((A1-A3).LT.0.0.AND.Q7.GT.0.0)GOTO 307
        DRO1P=-DRO1P
        COUNT=1.0
        IF((ROP1+DRO1P).LE.0.0.AND.ROP1.EQ.1E-5)GOTO 310
307     ROP1=ROP1+DRO1P
        IF(ROP1.LE.0.0)ROP1=1E-5
        IF(ROP1.EQ.1E-5.AND.COUNT.GE.6.0)ROP1=0.0
        IF(ROP1.EQ.0.0)GOTO 310
        IF(DRO1P.LE.2E-14.AND.DRO1P.GE.-2E-14)GOTO 310
        PRINT,ROP1,DRO1P,N,Z0
        GOTO 300
310     ROP2=RO
        COUNT=1.0
```

```
320   ROC=(N-1)*ROP2/(Z0-N+1)
      ROB=N*ROP2/(Z0-N)
      Q=8.0
      CALL HELP(N-1,M0,ROC,A1,M1,C,A21)
      CALL HELP(N,M0,ROB,A3,M1,B,A21)
      IF((C*ROC-B*ROB).EQ.0.0)PRINT302
      IF((C*ROC-B*ROB).EQ.0.0)ROP2=-1
      IF(ROP2.EQ.-1.0)GOTO 330
302   FORMAT('0ERROR #2')
      DRO2P=-(A1-A3)/((C*ROC-B*ROB)/ROP2)
      IF(COUNT.GT.2.0)GOTO 306
      IF(DRO2P.GT.0.0)COUNT=3.0
      IF(COUNT.EQ.3)GOTO 306
      IF(COUNT.EQ.1.0)DRO2P=-DRO2P
      IF(COUNT.EQ.1)Q=A1-A3
      IF(COUNT.EQ.1.0)GOTO 306
      IF((A1-A3).LT.0.0.AND.Q.GT.0.0)GOTO 306
      IF((A1-A3).GT.0.0.AND.Q.LT.0.0)GOTO 306
      DRO2P=-DRO2P
      COUNT=1.0
306   ROP2=ROP2+DRO2P
      IF(ROP2.GE.1E+04)ROP2=100
      IF(ROP2.EQ.100)GOTO 330
      IF(ROP2.LT.RO)ROP2=COUNT*RO
      IF(COUNT*RO.EQ.ROP2)COUNT=0.0
      IF(DRO2P.LE.2E-14.AND.DRO2P.GE.-2E-14)GOTO 330
```

```
      COUNT=COUNT+1
      PRINT,ROP2,DRO2P,N
      GOTO 320
330   PRINT340,ROP1,ROP2
340   FORMAT('0',D20.14,' <RO<',D20.14)
      PRINT900
900   FORMAT('-')
1000  CONTINUE
      STOP
      END
      SUBROUTINE NINE(NTOT,MO,P,PO,RO,NB,M1)
      DOUBLE PRECISION P(101),NTOT,PO,MO,RO,A,C,B,D,E,F,G,H,Z,Y,NB,M1
      DOUBLE PRECISION DLOG,DEXP
      ITEST=NTOT-MO+1
      I11=ITEST
      IF(I11.LT.1)I11=1
      DO 10 J1=1,ITEST
      J=J1-1
      Y=Z=0.0
      IF(ITEST.LT.1)GOTO 21
      DO 20 N1=1,ITEST
      NRESUP=N1-1
      A=(NRESUP-J)*DLOG(MO*RO)
      C=D=0.0
      IF(J.LE.1)GOTO 35
      DO 30 K=2,J
      B=K
```

```
30    C=C+DLOG(B)
35    IF(NRESUP.LE.1)GOTO 45
      DO 40 K=2,NRESUP
      B=K
40    D=D+DLOG(B)
45    E=A+C-D
      CALL FLOW(E,J,IX,IY)
      IF (E.LT.-174.996)GOTO 20
      Z=Z+DEXP(E)
20    CONTINUE
21    IF(ITEST.LE.1)ITEST=1
      ITOT=NTOT+1
      ITEST=ITEST+1
      DO 50 N1=ITEST,ITOT
      NRESUP=N1-1
      C=D=0.0
      M=MO
      IF(M.LE.1)GOTO 65
      DO 60 K=2,M
      B=K
60    C=C+DLOG(B)
65    IF(J.LE.1)GOTO 75
      DO 70 K=2,J
      B=K
70    D=D+DLOG(B)
75    A=F=0.0
      IF(NRESUP.LE.1)GOTO 85
```

```
      DO 80 K=2,NRESUP
      B=K
80    A=A+DLOG(B)
85    N=NTOT-NRESUP
      IF(N.LE.1)GOTO 95
      DO 90 K=2,N
      B=K
90    F=F+DLOG(B)
95    G=(NTOT-MO-J)*DLOG(MO)
      H=(NRESUP-J)*DLOG(RO)
      E=C+D+G+H-A-F
      CALL FLOW(E,J,IX,IY)
      IF(E.LT.-174.996)GOTO 50
      Y=Y+DEXP(E)
50    CONTINUE
      IF(J.EQ.0)PO=1.0/(Y+Z)
      IF(J.EQ.0)GOTO 10
      P(J)=1./(Y+Z)
10    CONTINUE
      CALL TEN(NTOT,MO,P,RO)
      CALL SIXTEN(NTOT,M1,P,NB,PO)
      RETURN
      END
      SUBROUTINE TEN(NTOT,MO,P,RO)
      DOUBLE PRECISION P(101),NTOT,PO,MO,RO,A,B,C,D,E,F,G,H,Z,Y
      DOUBLE PRECISION DLOG,DEXP
      INITAL=NTOT-MO+1
      IF(INITAL.LT.1)INITAL=1
```

```
      ITOT=NTOT
      DO 10 J=INITAL,ITOT
      Y=Z=0.0
      DO 20 N1=1,INITAL
      NRESUP=N1-1
      A=C=H=G=0.0
      IF(J.LE.1)GOTO 105
      DO 100 K=2,J
      B=K
100   G=G+DLOG(B)
105   N=NTOT-J
      IF(N.LE.1)GOTO 115
      DO 110 K=2,N
      B=K
110   H=H+DLOG(B)
115   IF(NRESUP.LE.1)GOTO 35
      DO 30 K=2,NRESUP
      B=K
30    A=A+DLOG(B)
35    N=MO
      IF(N.LE.1)GOTO 45
      DO 40 K=2,N
      B=K
40    C=C+DLOG(B)
45    D=(NRESUP-J)*DLOG(RO)
      E=(NRESUP-(NTOT-MO))*DLOG(MO)
      F=D+E+G+H-A-C
```

```
      CALL FLOW(F,J,IX,IY)
      IF (F.LT.-174.996)GOTO 20
      Y=Y+DEXP(F)
20    CONTINUE
      DO 50 NRESUP=INITAL,ITOT
      A=C=D=E=F=0.0
      IF(J.LE.1)GOTO65
      DO 60 K=2,J
      B=K
60    A=A+DLOG(B)
65    I=NTOT-J
      IF(I.LE.1)GOTO 75
      DO 70 K=2,I
      B=K
70    C=C+DLOG(B)
75    I=NTOT-NRESUP
      IF(I.LE.1)GOTO 85
      DO 80 K=2,I
      B=K
80    E=E+DLOG(B)
85    IF(NRESUP.LE.1)GOTO 95
      DO 90 K=2,NRESUP
      B=K
90    F=F+DLOG(B)
95    D=(NRESUP-J)*DLOG(RO)
      B=D+A+C-E-F
      CALL FLOW(B,J,IX,IY)
```

```
      IF(B.LT.-174.996)GOTO 50
      Z=Z+DEXP(B)
50    CONTINUE
      P(J)=1.0/(Y+Z)
10    CONTINUE
      RETURN
      END
      SUBROUTINE SIXTEN(NTOT,M1,P,NB,PO)
      DOUBLE PRECISION NTOT,M1,P(101),NB,PO,B
      ITOT=NTOT
      NB=0.
      INITAL=NTOT-M1+1
      IF(INITAL.LE.0)NB=PO
      IF(INITAL.LE.0)INITAL=1
      DO 10 NRESUP=INITAL,ITOT
      B=(M1-(NTOT-NRESUP))*P(NRESUP)
10    NB=NB+B
      RETURN
      END
      SUBROUTINE FLOW(X,J,IX,IY)
      DOUBLE PRECISION X
      IF(X.LT.-174.996)IX=IX+1
      IF(X.GT.174.996)IY=IY+1
      IF(X.LT.-174.996)PRINT5,J
5     FORMAT(' *****UNDERFLOW*****   J=',I3)
      IF(X.GT.174.996)PRINT10,J
```

```
10    FORMAT(' *****OVERFLOW*****   J=',I3)
      IF(X.GT.174.996)X=174.99
      RETURN
      END
      SUBROUTINE NBSTK(NB,NTOT,MO,M1,RO)
      DOUBLE PRECISION NTOT,M1,NB,STOCK,D,C,MO,RO,PO
      DOUBLE PRECISION DLOG,DEXP
      M=NTOT-M1+1
      ITEST=100
      NB=0.0
      IF(M.LE.0)NB=DEXP(-MO*RO)
      IF(M.LE.0)M=1
      DO 50 NRESUP=M,ITEST
      C=0.0
      IF(NRESUP.LE.1)GOTO 65
      DO 60 J=2,NRESUP
      D=J
60    C=C+DLOG(D)
65    IF(NRESUP*DLOG(MO*RO)-C-MO*RO.LT.-100)GOTO 111
50    NB=NB+(M1-NTOT+NRESUP)*DEXP(NRESUP*DLOG(MO*RO)-C)*DEXP(-MO*RO)
111   CONTINUE
      RETURN
      END
      SUBROUTINE HELP(N,MO,ROB,A3,M1,B,A21)
      DOUBLE PRECISION N,MO,ROB,A3,M1,B,A2,P(101),PO,DER1,A21
      IF(A21.EQ.0.0)CALL NBSTK(A2,N-1,MO,M1,ROB)
      IF(A21.EQ.0.0)CALL NBSTK(A3,N,MO,M1,ROB)
```

```
      IF(A21.EQ.0.0)DER1=MO*ROB
      IF(A21.NE.0.0)CALL NINE(N-1,MO,P,PO,ROB,A2,M1)
      IF(A21.NE.0.0)CALL NINE(N,MO,P,PO,ROB,A3,M1)
      IF(A21.NE.0.0)CALL SUB21(N,P,DER1)
      B=-(DER1/ROB)*(A3-A2)
      RETURN
      END
      SUBROUTINE SUB21(N,P,DER1)
      DOUBLE PRECISION N,P(101),DER1
      DOUBLE PRECISION DLOG,DEXP
      DER1=0.0
      ITOT=N
      DO 10 NRESUP=1,ITOT
10    DER1=DER1+NRESUP*P(NRESUP)
      RETURN
      END
$ENTRY
.5        4        6
/LOGOFF
```

APPENDIX B

EXAMPLE NUMERICAL RESULTS

EXAMPLE PROBLEM 2 RESULTS

INPUT DATA				OUTPUT DATA				
ρ_0	$m_1 = m_0$	z_0	(0=Poisson, 1=parallel)	N_{opt}	$[N_B]_{opt}$	$(\rho_1)_{opt}$	$(\rho_0)_{min}$	$(\rho_0)_{max}$
0.01	1	5.5	0	4	0.20736×10^{-7}	0.02667	0.25614×10^{-2}	0.15996
			1	4	0.20516×10^{-7}	0.02667	0.25766×10^{-2}	0.17726
0.5	1	5.5	0	3	0.02691	0.60000	0.15996	0.76264
			1	3	0.01982	0.60000	0.17726	1.12582
1.00	1	5.5	0	2	0.13615	0.57143	0.76264	1.89260
			1	3	0.08978	1.20000	0.17726	1.12582
0.01	2	7.5	0	6	0.25891×10^{-6}	0.04000	0.41046×10^{-3}	0.04444
			1	6	0.25543×10^{-6}	0.04000	0.41090×10^{-3}	0.04584
0.5	2	7.5	0	4	0.14513	0.57143	0.25339	0.69908
			1	4	0.10384	0.57143	0.28761	1.05893
1.00	2	7.5	0	3	0.59693	0.66667	0.69908	1.39217
			1	4	0.36983	1.14286	0.28761	1.05893
0.01	5	12.5	0	10	0.77095×10^{-6}	0.04000	0.005398	0.04159
			1	10	0.76179×10^{-6}	0.04000	0.005421	0.04251
0.5	5	12.5	0	7	1.39692	0.63636	0.29457	0.50919
			1	7	0.88350	0.63636	0.36448	0.79758
1.00	5	12.5	0	5	3.33333	0.66667	0.75621	1.02000
			1	6	1.89229	0.92308	0.79758	1.60782
0.01	10	25	0	21	0.58778×10^{-12}	0.05250	0.0034220	0.01329
			1	21	0.58266×10^{-12}	0.05250	0.0034252	0.01333
0.5	10	25	0	13	2.54328	0.54167	0.49710	0.60890
			1	14	1.59840	0.63636	0.48681	0.70466
1.00	10	25	0	9	6.62500	0.56250	0.96000	1.08129
			1	12	3.76301	0.92308	0.99203	1.37691
0.01	15	35	0	29	0.32645×10^{-14}	0.04833	0.007865	0.01793
			1	29	0.32428×10^{-14}	0.04833	0.007872	0.01891
1.00	15	35	0	12	10.82528	0.52174	0.96297	1.04968
			1	17	6.25727	0.94444	0.99947	1.25728
0.5	20	40	0	20	10.00000	0.50000	0.47500	0.52500
			1	23	6.28137	0.67647	0.49201	0.60460

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER AFOSR-TR- 78-0904	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) ANALYSIS OF INVENTORY/RESUPPLY SUBSYSTEM MODELS IN OPTIMAL MATERIEL DISTRIBUTION SYSTEM DESIGN		5. TYPE OF REPORT & PERIOD COVERED Final
		6. PERFORMING ORG. REPORT NUMBER UDR-TR-78-40;UDSE-TR-78-04
7. AUTHOR(s) Arthur Z. Kovacs		8. CONTRACT OR GRANT NUMBER(s) AFOSR 77-3266
9. PERFORMING ORGANIZATION NAME AND ADDRESS University of Dayton School of Engineering Dayton, Ohio 45469		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 61102F 2304/D9
11. CONTROLLING OFFICE NAME AND ADDRESS Air Force Office of Scientific Research/NM Bolling AFB, DC 20332		12. REPORT DATE April 1978
		13. NUMBER OF PAGES 153
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

backorder level, conservative system, demand process modeling, gradient technique, inventory modeling, investment items, logistics system, optimal investment level, optimal resupply time level, resupply system modeling, queueing models

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The purpose of the Air Force freight distribution system is to meet spare parts demand requirements at minimum cost. Budgetary constraints have suggested that total expected backorder level for items at user installations be minimized subject to a given dollar expenditure level for inventory investment. LOGAIR (a dedicated Air Force air transport service) is a major transportation sub-system to support spare parts delivery requirements of users of high priority items. A two-echelon inventory system for spare parts

20. Abstract

delivery exists with centralized, specialized inventories at the Air Logistics Centers (ALCs) and decentralized, broad profile inventories located at user installations. LOGAIR provides transport with low order and ship time to reduce resupply time in the maintenance of inventory safety levels at user bases. A systems approach is used to formulate a cost/benefit model which recognizes the impact of the Air Force resupply system upon the total spare parts distribution system in terms of total inventory investment level, total system cost, and backorder level. Given a total expenditure level available for allocation between inventory investment and transportation, the problem is to determine the optimal fractional allocation to be made to transportation (the remaining fraction to be allocated to inventory investment) such that total expected backorder level is minimized. A conceptual framework for trade-off analysis for minimizing total system cost in terms of inventory investment level and resupply time level for a given backorder level is presented. This conceptual framework also allows for the minimization of backorder level in terms of inventory investment level and resupply time level for a given total dollar expenditure level. Queueing models exhibiting various demand and resupply processes are explored and compared to determine the impact of inventory investment level and resupply time level upon backorder level. Specific solution procedures are developed for and are applied to the trade-off analyses mentioned above.